
APRIL 1977 VOLUME 2, Number 4
$1.50
BYTE
the small systems journal

Meet the most powerful μC system available for dedicated work. Yet it's only $595.*

Here's the muscle you've been telling us you wanted: a powerful Cromemco microcomputer in a style and price range ideal for your dedicated computer jobs — ideal for industrial, business, instrumentation and similar applications.

It's the new Cromemco Z-2 Computer System. In the Z-2 you get all of the following for only $595:

- The industry's fastest μP board (4 MHz or 250-nanosecond cycle time).
- The power and convenience of the well-known Z-80 μP.
- A power supply you won't believe (+8V @ 30A, +18V and −18V @ 15A. That's not only ample power for a full set of cards but ample additional power for other peripherals such as floppy disk drives).
- A full-length **shielded** motherboard with 21 card slots to let you plug in almost any conceivable combination of memory, I/O or your own custom circuits.
- Power-on-jump circuitry to begin automatic program execution when power is turned on.
- The S-100 bus that's widely supported by a host of peripherals manufacturers.
- All-metal chassis and dust case.
- Standard rack-mount style construction that's usable with a variety of cabinets and is upward compatible with larger systems. Bench cabinet optional.
- Card retainer to secure cards in sockets under all conditions.
- 110- or 220-volt operation.

DEDICATED APPLICATIONS

The new Z-2 is specifically designed as a powerful but economical dedicated computer for systems work. Notice that the front panel is entirely free of controls or switches of any kind. That makes the Z-2 virtually tamper-proof. No accidental program changes or surprise memory erasures.

(For your custom work, Cromemco will supply blank panels.)

4 MHz SPEED

Cromemco's microcomputers are the fastest and most powerful available. The 4 MHz speed of the Z-2 means you get about twice the speed of nearly any other microcomputer.

(continued on next page)

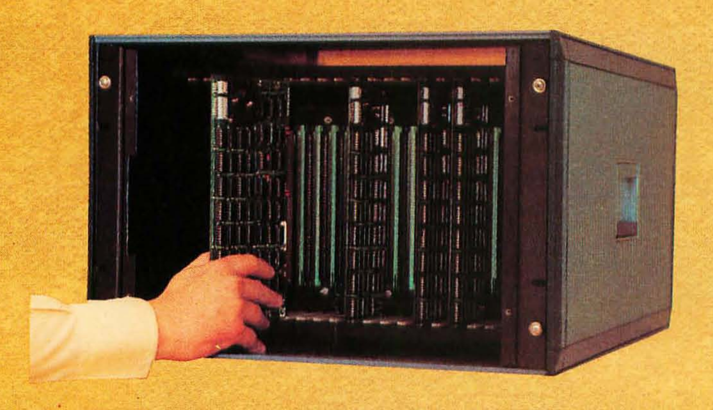


Shown with optional bench cabinet

*kit price



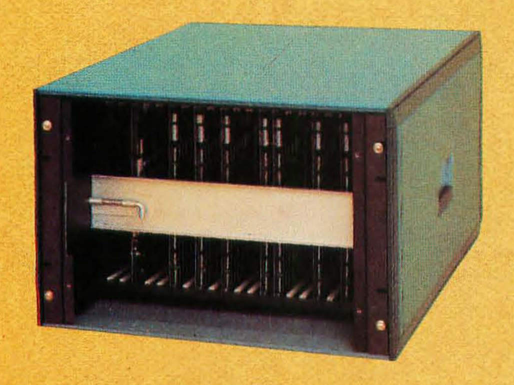




Full-width motherboard has slots for 21 cards.



Z-2 is supplied for rack mounting. Attractive bench cabinet shown is also available.



Heavy-duty retainer bar insures that cards won't jostle out of sockets.

That 4 MHz speed cuts running time in half — cuts your waiting time in half. Lets you do more sophisticated work that might not otherwise be practical with a microcomputer.

Z-80 GIVES MORE COMPUTER POWER

The Z-2 uses the powerful Z-80 microprocessor with its 158-instruction set, 19 internal registers, 10 addressing modes, and 16 bit arithmetic operations.

The Z-80 is widely regarded as the standard microprocessor of the future. So you're in the technical fore with the Z-2. But you can also plug in other microprocessor boards if you wish.

BROAD SOFTWARE SUPPORT

Since the Z-2 uses the Z-80, your present 8080 software can be used with the Z-2. In addition, Cromemco also offers software support including a monitor, assembler, and a BASIC interpreter.

STANDARD S-100 BUS

The Z-2 uses the S-100 bus that has become standard in the microcomputer field. Dozens of manufacturers support it with compatible peripherals.

Of course, all Cromemco peripherals such as our 7-channel A/D and D/A converter, our well-known BYTESAVER with its built-in PROM programmer, our color graphics interface, etc., will also plug into the bus.

LOW NOISE

Cromemco engineers have been careful to design the Z-2 for reliable operation at its fast 4 MHz speed. Noise on the motherboard, for example, has been carefully minimized by a unique ground-plane design (we call it our 'Blitz Bus™'). With this bus you won't get erratic operation due to ground current noise.

RACK OR CABINET MOUNTING

The basic Z-2 is supplied in a black-anodized metal case for mounting in a standard 19-inch relay rack. A high-quality stylized bench cabinet in an attractive blue color is also available.

KIT OR ASSEMBLED

You can get the Z-2 either kit or assembled. The kit includes the Z-2 for rack mounting, the Cromemco 4 MHz microprocessor card, full-length 21-card-slot motherboard, power supply, one card socket and card-guide set, and front panel.

The assembled Z-2 includes the above as well as all 21 sockets and card guides and a cooling fan.

LOW, LOW PRICE

All who've seen the Z-2 have been surprised and impressed with its low price.

You'll be impressed, too, with its technical excellence and quality.

So see it right away at your computer store — or order now directly from the factory.

The sooner you put this advanced computer to work for you, the sooner you'll be solving problems and saving time/money.

Z-2 Computer System kit (Model Z-2K) ..... $595.
Z-2 Computer System assembled (Model Z-2W) .... $995.



Cromemco
i n c o r p o r a t e d
Specialists in computers and peripherals
2432 CHARLESTON RD., MOUNTAIN VIEW, CA 94043 • (415) 964-7400

In the Queue

BYTE APRIL 1977
Volume 2
Number 4

Foreground

Background

Nucleus

BYTE magazine is published monthly by BYTE Publications, Inc, 70 Main St, Peterborough, New Hampshire 03458. Subscription rates are $12 for one year, $22 for two years, $30 for three years in North America. Second class postage paid at Peterborough New Hampshire 03458 and at additional mailing offices. Phone 603–924-7217. Address editorial correspondence to Editor, BYTE, 70 Main St, Peterborough NH 03458. Opinions expressed by authors are not necessarily those of BYTE magazine.

Address all subscription correspondence to:
BYTE Subscriptions
PO Box 361
Arlington MA 02174
617-646-4329

In This BYTE

Many people who play with computers have seen a version of that well known game, "lunar lander." Much has been written about different versions of the game in books and other publications. In his article, **KIM Goes to the Moon**, Jim Butterfield adds one more chapter as he describes not only a version of the game, but also the logic behind his particular development of a lunar lander program to fit the limited resources of a single board computer.

Are you thinking of adding a hardcopy device to your microprocessor but can't afford the price that is required? Well, there are a lot of old Baudot type teleprinters out there which have plenty of use left in them. Michael McNatt describes some of these devices that are available on the surplus market today in his article, **A Guide to Baudot Machines: Part 1, Description of Available Devices.**

Watch out for complicated interpersonal situations. Your computer, if it is found out, could lead to events reminiscent of a jai alai fronton at mid game, as you'll find out by reading Steve Ciarcia's humorous account of **Having a Private Affair With Your Computer.**

Tom Pittman's Tiny BASIC language provides both the novice and experienced programmer with a vehicle for conveying thoughts from the flowchart stage to working programs quickly and efficiently. Tiny BASIC is a language that can be quickly mastered by the novice, yet has enough variations to satisfy the experienced programmer. In the article **A Review of Tom Pittman's Tiny BASIC**, Richard Rosner details what abilities the language provides. He also gives a fine example of the type of programming which can be accomplished in one short evening using such a high level language.

Waiting for a slow audio tape interface is one of the annoying aspects of stand alone computer systems without much in the way of mass storage peripherals. In this issue, find out how you can run a Southwest Technical Products Corporation AC-30 tape interface at four times its usual rate in the article on **A Software Controlled 1200 bps Audio Tape Interface.** This same interface can incidentally be run at 300 bps to read Kansas City (BYTE) standard tapes for which the AC-30 was intended.

For some time there has been a mystique associated with the phrase "artificial intelligence." The mystery often lies in defining what it is. If we take the definition offered by Turing as "able to mimic the behaviour and decision making of a human," then we are using artificial intelligence every time we play a simple computer game such as nim. In his article **Artificial Intelligence, What Is It?**, Richard L Rosenbaum dispels some of the mystique that surrounds artificial intelligence, by means of some introductory background information.

No system is truly complete without some systems software to help you create personalized software applications. But if memory is limited, how can you accomplish minimal functions such as assembly of programs? One way for 6800 users with 4 K bytes or more is described in Jack Emmerichs' article on **Designing the "Tiny Assembler."** In this first part, he describes the necessary prelude to such a design: **Defining the Problem.** (In our next issue, the article continues with detailed information, object code and some comments on customization to individual circumstances.)

Learn about **Establishing the CHU Dynasty** in your local computer club by reading Steven B Gray's somewhat tongue-in-cheek article . . .

About the Cover . . .

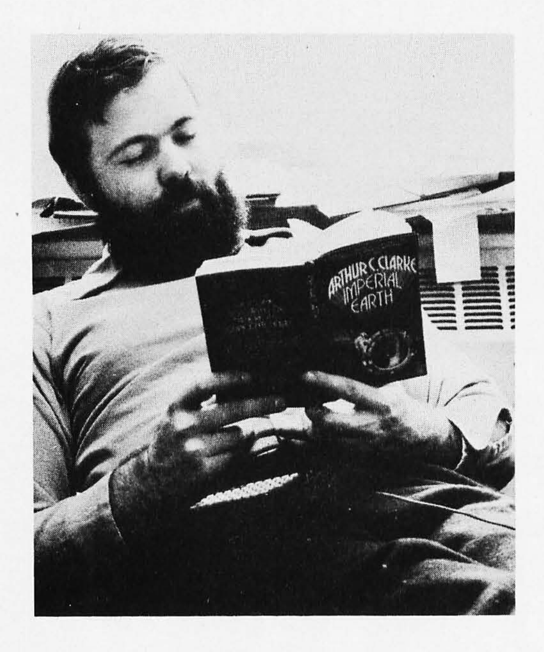
April Fool! This cover depicts a working (believe it or not) computer called "Spider," which was first seen in a black and white shot in our August 1976 issue. The computer was built by Roger Amidon, and this color print was taken by Marj Kirk. For those who missed the August issue, the method of construction for this 12 bit mini was to wire point to point between small circuit cards. The Niagara Falls effect of spilling off the table was due to a wayward cat.

Who is Plexitus? He's the main character in E E Barnes' hysterical account of a jinxed Roman named Plexitus who is credited with the invention of flight, creation of the word idiot, and last (but hardly least) the invention of the computer. **Early Indications of Technology in Roman Military Arts or Plexitus** is a most interesting fictional history.

Science fiction has long been the mainstay of people who are interested in the frontiers of the future, the possibilities of tomorrow's technology today. But living in the world of today has made much of yesterday's science fiction ordinary occurrence. And some of today's technology hasn't even been touched upon by science fiction writers! Henry Melton's essay, **Why Aren't There Any Altairs on Arcturus II?**, concerns the curious absence of personal computers in the works of science fiction, a small discrepancy in the vision of our prophets . . .

In this issue, we conclude the three part series about Omega **Navigation with Mini-O** with Richard J Salter's article on the software used to drive the hardware described in Ralph Burhans' articles. In Richard's article you'll find out how to use a 6502 processor to measure Omega phase differences, and as an extra bonus, how to calibrate a local clock in your lab with the Cesium atomic clocks used by the Omega system.

Editorial



How I Was Born 300 Years Ahead of My Time

by Carl Helmers

It started late last summer, the first of the rumors and innuendos. Someone walked up to me and said words to the effect of "you know, Carl, Arthur C Clarke writes well about you. . . ." How would you feel if someone walked up to you and made such a profound statement? To an old science fiction addict turned science fictionish reality purveyor, it is a rather astonishing statement. Arthur C Clarke is sitting up there in my private pantheon of the gods of great literature along with Robert Heinlein, Poul Anderson, Isaac Asimov, and Larry Niven (as well as editor John Campbell, now deceased, who cultivated the talents of many science fiction writers over the years). I've never spoken to the man, communicated with him, let alone imagined that I would. He's the person who suggested the idea of the synchronous satellite before today's technology ever existed, and is author of numerous widely read science fiction stories. So when someone suggested to me that one of those remote names on book jackets had somehow taken notice of yours truly, my immediate reaction was to file it as amusing and forget it, as if someone had walked up to me and said "I understand Herman Melville had a whale of a good time with you when you guys went to school together."

However, this vicious rumor kept dogging me. More than one person mentioned it in passing. Eventually I found out, another example of the truism that if there is a persistently rumored or mentioned fact, there is often a source to that story, whether it be a persistent gossip columnist or a more pleasing manifestation of reality. It seems that according to Arthur C Clarke, I really lived in the 23rd century of our current calender: That noted historian of the future wrote a historical novel about my adventures as the mildly tragic hero of his latest book on life in the 23rd century, *Imperial Earth*. The book is available at any local book store in paperback form, on the science fiction shelves. It's Clarke's contribution to the American Bicentennial. Like most historical novels (according to my mental stereotype), it is a mildly boring book which takes a lot of pages and features a slowly developing plot, giving ample opportunity to tour the wonders of the age in question. I find on reading the book that one of the three major characters in the story is named (at various places – the editors were inconsistent) Karl Helmer or Karl Helmers.

What a shock. While I would have liked to imagine myself the principal by namesake of an Ayn Rand novel, I guess I'll just have to put up with Arthur C Clarke as a distant second best. (I really have no choice in the matter, and I'm certainly not about to be so superficial as to change my name to John Galt.) However, it is an interesting coincidence since there are numerous parallels with the character in the story (as well as vast differences). Here I am, fancying myself an imaginative technologist in an age of technology, revelling in the fun I'm having, and all of a sudden I find that the hero of one of my own hero's novels is an imaginative technologist in another era, with a very close approximation to my own name. Aside from the unfortunate end of the hero's life in the story, the net effect of reading the novel was an ego boost I could not in all modesty ignore.

Personal Computing in 2276

Science fiction occasionally catches up with reality. (See Henry Melton's article "Why Aren't There Any Altairs on Arcturus IV?" in this issue.) Arthur C Clarke's *Imperial Earth* presents a concept of the personal computer which is in effect an evolutionary outgrowth of what we see now, but takes the pocket calculator (programmable) as his point of departure. The name of the beast is the "Minisec," a sort of advanced pocket calculator, which is introduced in chapter 18 when the principal character of the story (Duncan Makenzie) reaches Earth on his pilgrimage to the 2276 celebration of the Quincentennial. The specifications of the Minisec? Read the following excerpt from

Continued on page 133

Articles Policy

BYTE is continually seeking quality manuscripts written by individuals who are applying personal systems, or who have knowledge which will prove useful to our readers. Manuscripts should have double spaced typewritten texts with wide margins. Numbering sequences should be maintained separately for figures, tables, photos and listings. Figures and tables should be provided on separate sheets of paper. Photos of technical subjects should be taken with uniform lighting, sharp focus and should be supplied in the form of clear glossy black and white or color prints (if you do not have access to quality photography, items to be photographed can be shipped to us in many cases). Computer listings should be supplied using the darkest ribbons possible on new (not recycled) blank white computer forms or bond paper. Where possible, we would like authors to supply a short statement about their background and experience.

Articles which are accepted are typically acknowledged with a binder check 4 to 8 weeks after receipt. Honorariums for articles are based upon the technical quality and suitability for BYTE's readership and are typically $25 to $50 per typeset magazine page. We recommend that authors record their name and address information redundantly on materials submitted, and that a return envelope with postage be supplied in the event the article is not accepted. ■

Experience the excitement of owning the finest *personal* computer – IMSAI 8080

Waiting for you — all the incredible performance and power of the IMSAI 8080. And at a price you would normally pay for a home music system: $699 in kit; $931 assembled.

WORLD OF USES

The IMSAI 8080 is a superbly engineered, quality computer. It is versatile, expandable and powerful, putting literally hundreds of applications and uses at your fingertips. Imagine sitting at your desk and enjoying interaction with your IMSAI 8080! Press the *on* switch and you're ready for game playing, research, education, business applications, or for laboratory instrument control. It has all the power you need, and more, to make your application ideas come alive.

GROWS WITH YOU

The IMSAI 8080 is designed for many years of pleasure. With its open-chassis engineering you can expand your system by adding peripherals and interfaces. The 22-slots and 28 amp power supply mean that you can plug-in today's, plus *tomorrow's* modules.

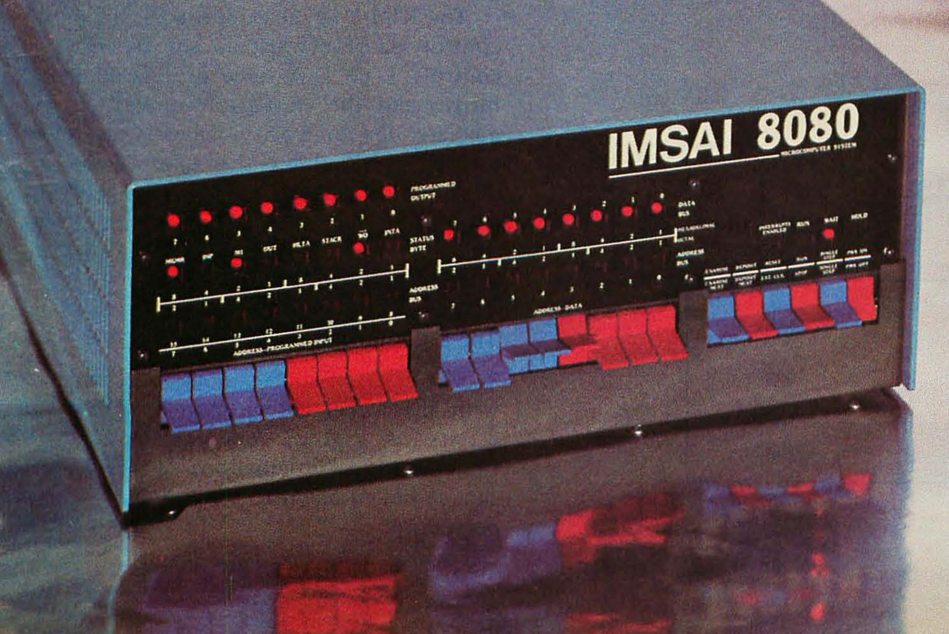
Right now you can add a module for displaying color graphics and characters on TV; a ready-to-use keyboard; small and large printers, and a single interface that lets you attach multiple devices including a cassette tape recorder. Expect the latest, exciting equipment from IMSAI. We are committed to leadership in this expanding technology.

EASY TO PROGRAM

With our BASIC language you can operate the IMSAI 8080 quickly and easily. Technically knowledgeable? Use our assembly language to develop sophisticated and unique software.

If you're thinking personal computer, treat yourself to the very best – IMSAI 8080

Send for free four-color brochure or $1 for catalog. Call or write for name of nearest dealer.



IMSAI

IMSAI Manufacturing Corporation
14860 Wicks Blvd.
San Leandro, CA 94577
(415) 483-2093
TWX 910-366 7287

Prices: USA Domestic

KIM Goes to the Moon

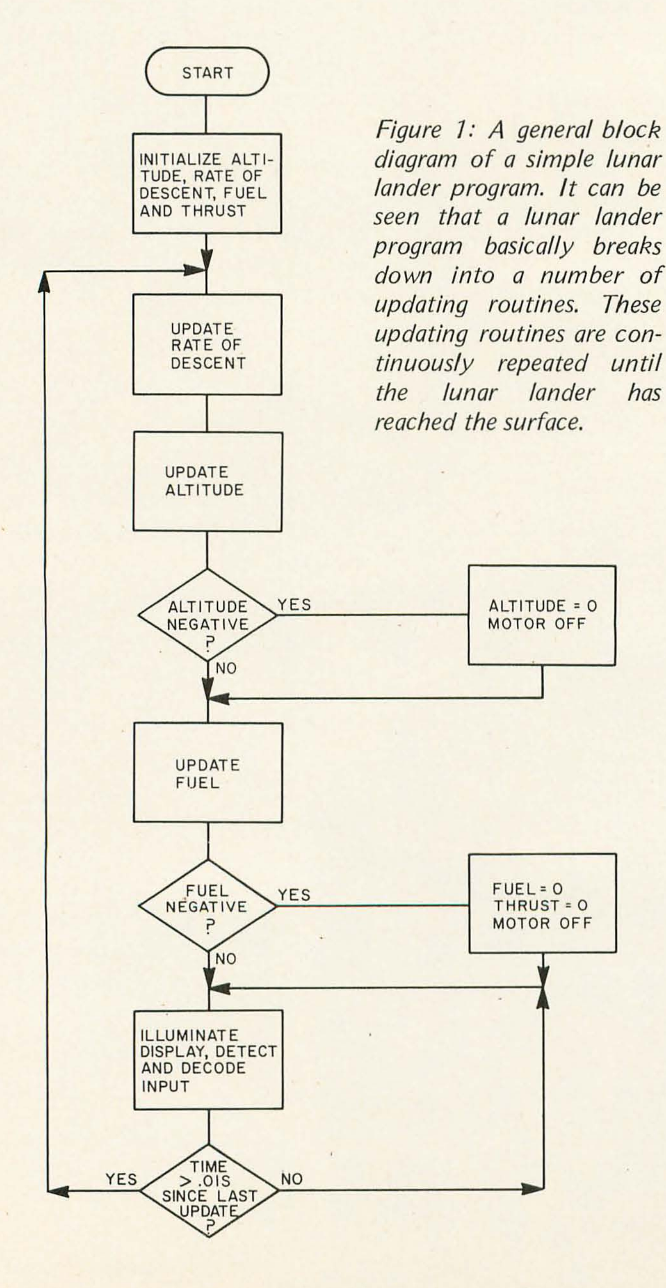


Figure 1: A general block diagram of a simple lunar lander program. It can be seen that a lunar lander program basically breaks down into a number of updating routines. These updating routines are continuously repeated until the lunar lander has reached the surface.

Jim Butterfield
14 Brooklyn Av
Toronto Ontario M4M 2X5 CANADA

There are quite a few lunar landing programs available nowadays: some for pocket calculators, others using graphic displays. The one I wrote for my KIM-1, based on the MOS Technology 6502 microprocessor, illustrates many of the techniques needed to develop the program.

The KIM-1 comes with a six digit LED display, which can be accessed by the user. I used the first four digits to represent the craft's altitude, and optionally, the fuel remaining. The last two digits, which are slightly separated from the rest of the display, are used for rate of descent. Both values change continually as the craft moves.

The KIM-1 keyboard is used as the pilot's control panel. Thrust is set by pressing controls 1 to 9. A value of 1 is minimum thrust, and the craft's rate of descent will increase due to gravity. Nine is maximum thrust, which slows the rate of descent sharply. In addition to power control, the pilot can elect to view either current altitude, by pressing A, or remaining fuel, by pressing F.

The Equations of Motion

The craft, of course, moves in accordance with the forces acting upon it: thrust and gravity. A physics textbook shows some rather formidable equations. However, they can be boiled down to the following simple procedure:

Every 0.01 second,

- add 0.01 of the acceleration to the velocity;
- add 0.01 of the velocity to the altitude;
- subtract 0.01 of the thrust from the fuel.

The acceleration is set equal to thrust minus gravity, and gravity is set at the constant value 5.

The time period of 0.01 s is arbitrary. Since KIM can operate in decimal mode, dividing by 100 becomes an elementary operation. Everything would work just as well if it were done in any other small time increment.

Figure 1 shows an elementary block diagram of the program. After setting the initial flight values, we settle into three main jobs: updating the flight, lighting the display, and detecting input from the pilot.

Setting Initial Values

An interesting flight can be obtained by starting the lunar module at a height of 4,500 feet with 800 pounds of fuel. That's more than sufficient fuel for a safe landing, but not enough to allow for prolonged hovering.

It's not difficult to set all the initial values by programming them individually. However, a faster method is to set them all together in memory and use a loop to initialize all of them. This is what I did as shown in listing 1 on hexadecimal lines 0000 to 0007.

Updating the Flight

Every 0.01 s we must update our rate of descent, altitude and fuel. As previously indicated, we have to add 0.01 of various values into the totals. We can accomplish this quite easily by using a gimmick. Instead of holding the altitude, for example, in feet, let's use two more digits and store it as multiples of 0.01 feet. Now we can add the rate of ascent directly into the six digit number; and the division by 100 happens automatically. For display purposes, of course, we drop the last two digits, so that we're back to height in feet. Using the same technique on the other parameters, we find that the updating job becomes relatively easy.

During the updating task, we must also detect two special conditions: touchdown and out of fuel. This seems fairly simple

Text, listing continued on page 132

Listing 1: An example lunar lander program written for the KIM-1 microprocessor that uses the flowchart of figure 1 as a base. The input and output of this program is handled by routines that are inherent to the KIM-1 system. The data display is seen on the keypad and LED display of the KIM-1 assembly. This display continuously shows the rate of descent, and on command will display either the amount of fuel left, or the altitude of the craft. Keys 1 through 9 are used to input thrust commands, while key A chooses the altitude display mode and the F key chooses the fuel display mode. All the numbers in this listing are in hexadecimal unless otherwise stated.

Address	Op	Operand	Label	Mnemonic	Commentary
0000	A2	0C	GO	LDX #0C	
0002	B5	B8	LP1	LDA INIT,X	initialize values;
0004	95	E2		STA ALT,X	
0006	CA			DEX	
0007	10	F9		BPL LP1	
0009	A2	05	CALC	LDX #05	X:=05;
000B	A0	01	RECAL	LDY #01	Y:=01;
000D	F8			SED	set decimal mode;
000E	18			CLC	clear carry;
000F	B5	E2	DIGIT	LDA ALT,X	
0011	75	E4		ADC ALT+2,X	add each digit;
0013	95	E2		STA ALT,X	
0015	CA			DEX	
0016	88			DEY	set up next digit;
0017	10	F6		BPL DIGIT	
0019	B5	E5		LDA ALT+3,X	
001B	10	02		BPL INCR	
001D	A9	99		LDA #99	
001F	75	E2	INCR	ADC ALT,X	
0021	95	E2		STA ALT,X	
0023	CA			DEX	counter:=counter - 1;
0024	10	E5		BPL RECAL	if counter positive go to RECAL;
0026	A5	E2		LDA ALT	else check if altitude is positive;
0028	10	0B		BPL UP	if altitude positive go to UP;
002A	A9	00		LDA #00	else altitude:=00;
002C	A2	02		LDX #02	X:=02
002E	95	E2	DD	STA ALT,X	
0030	95	E8		STA TH2,X	else turn off engine;
0032	CA			DEX	
0033	10	F9		BPL DD	
0035	38		UP	SEC	set carry;
0036	A5	ED		LDA FUEL+2	
0038	E5	EA		SBC THRUST	update fuel;
003A	85	ED		STA FUEL+2	
003C	A2	01		LDX #01	
003E	B5	EB	LP2	LDA FUEL,X	
0040	E9	00		SBC #00	
0042	95	EB		STA FUEL,X	check if fuel left;
0044	CA			DEX	
0045	10	F7		BPL LP2	
0047	B0	0C		BCS TANK	if fuel left go to TANK;
0049	A9	00		LDA #00	
004B	A2	03		LDX #03	
004D	95	EA	LP3	STA THRUST,X	else turn off engine;
004F	CA			DEX	
0050	10	FB		BPL LP3	
0052	20	AA 00		JSR THRSET	go to THRSET;
0055	A5	EE	TANK	LDA MODE	A:=display mode;
0057	D0	0A		BNE SHOFL	if mode not 00 go to SHOFL;
0059	A5	E2		LDA ALT	AX:=location of altitude;
005B	A6	E3		LDX ALT+1	
005D	F0	08		BEQ ST	go to ST;
005F	D0	06		BNE ST	
0061	F0	A6	LINK	BEQ CALC	
0063	A5	EB	SHOFL	LDA FUEL	A:=FUEL;
0065	A6	EC		LDX FUEL+1	X:=FUEL+1;
0067	85	FB	ST	STA POINTH	display values;
0069	86	FA		STX POINTL	
006B	A5	E5		LDA VEL	A:=velocity sign;
006D	30	06		BMI DOWN	if sign negative go to DOWN;
006F	A5	E6		LDA VEL+1	A:=/velocity/;
0071	F0	07		BEQ FLY	go to FLY;
0073	D0	05		BNE FLY	
0075	38		DOWN	SEC	
0076	A9	00		LDA #00	velocity:=/velocity/;
0078	E5	E6		SBC VEL+1	

A Nybble on the Apple



Notes by Carl Helmers

Next month, we'll have an article by Steve Wozniac, designer of the Apple-II computer, describing this beautiful new conception of the small computer. As a taste of the Apple-II, here is one frame of the "Color Eater," an interesting program which illustrates the facilities of the Apple-II.

The Apple-II, which is to be introduced in April at the first West Coast Computer Faire in San Francisco, may be the first product to fully qualify as the "appliance computer." An "appliance computer" is by definition a completed system which is purchased off the retail shelf, taken home, plugged in and used. I first saw the Apple-II on November 20 1976 when Stephen Wozniak and Stephen Jobs stopped by a motel room in Palo Alto where I was staying at the time. They brought along the prototype Apple-II to give a demonstration.

To serve as an interesting challenge to its capabilities, I suggested that we sit down and implement a "Color Eater" algorithm with Apple-II's integer BASIC interpreter with color graphics extensions. I had first seen the Color Eater program demonstrated in an advanced graphics research laboratory late in 1975 (the idea of the program is not original with me, and I will provide the source upon request). The Color Eater always lives in the matrix in the color TV display at some point.

The Color Eater is a very simple animal. It looks at its nearest neighbors in the color matrix, searching in a clockwise direction for its current "digestible" color. If it finds this color, it moves its location to the matrix position of that color, digests it into a new color, and reiterates its search. Occasionally, the Color Eater becomes a very frustrated little animal. It eats itself into a corner and no longer is able to find any digestible colors. When this catastrophe happens, it throws a fit and turns itself into another variety of Color Eater which can eat itself out of the frustration point. The result is a constantly changing random color pattern on the screen, illustrated in one state in this photo made with the Apple-II computer's output to a standard color television.

That evening last November, Steve Jobs, Steve Wozniak and I sat down and proceeded to use the Apple-II BASIC (which is a 5 K interpreter with 16 bit integer arithmetic) to program the Color Eater game. After perhaps 30 to 45 minutes, we had a working BASIC language version which used the Apple-II's graphics facilities. Since it was done interpretively, the program ran a bit slow in this version, but it certainly illustrated the concept. Later, Steve Wozniak recoded the program using the 6502 processor's assembly language facility as implemented in the Apple-II, and reports that the Color Eater now runs like lightning, which is its normal mode of operation these days as a demonstration program for the Apple-II.

If you attend the first West Coast Computer Faire, stop by the Apple Computer booth and take a look at this interesting processor; if you don't see it in person, then you'll have to wait until next month's BYTE for a more complete description of the design concepts of the machine as explained by Steve Wozniak.■

ANNOUNCING OSI 6502 8K BASIC.

OSI's new 8K BASIC for the 6502 was written by Microsoft, the people who wrote ALTAIR® 8K BASIC for the 8080. OSI's 6502 8K BASIC is identical to this powerful and popular 8K BASIC with two very important exceptions: our OSI 6502 8K BASIC has automatic string space handling, and it runs faster. Up to 8 times faster than the 8080 BASIC. And hundreds of times faster than many 6800 BASICs.

In fact, the OSI Challenger with OSI 6502 8K BASIC can actually outperform most small- and medium-scale minicomputers, as well as every micro there is! And that includes the Z-80.

Perhaps even more amazing than its superlative performance is its surprisingly low price: either $50 or free.

OSI 6502 8K BASIC is available to OSI System kit builders for $50 on your choice of paper tape, audio cassette or floppy disk.

And OSI 6502 8K BASIC comes free with the purchase of any 12K or larger OSI Challenger.

So you can own a fully-assembled OSI Challenger complete with serial interface, 12K of RAM memory and our OSI 6502 8K BASIC for just $807.

Incredible!

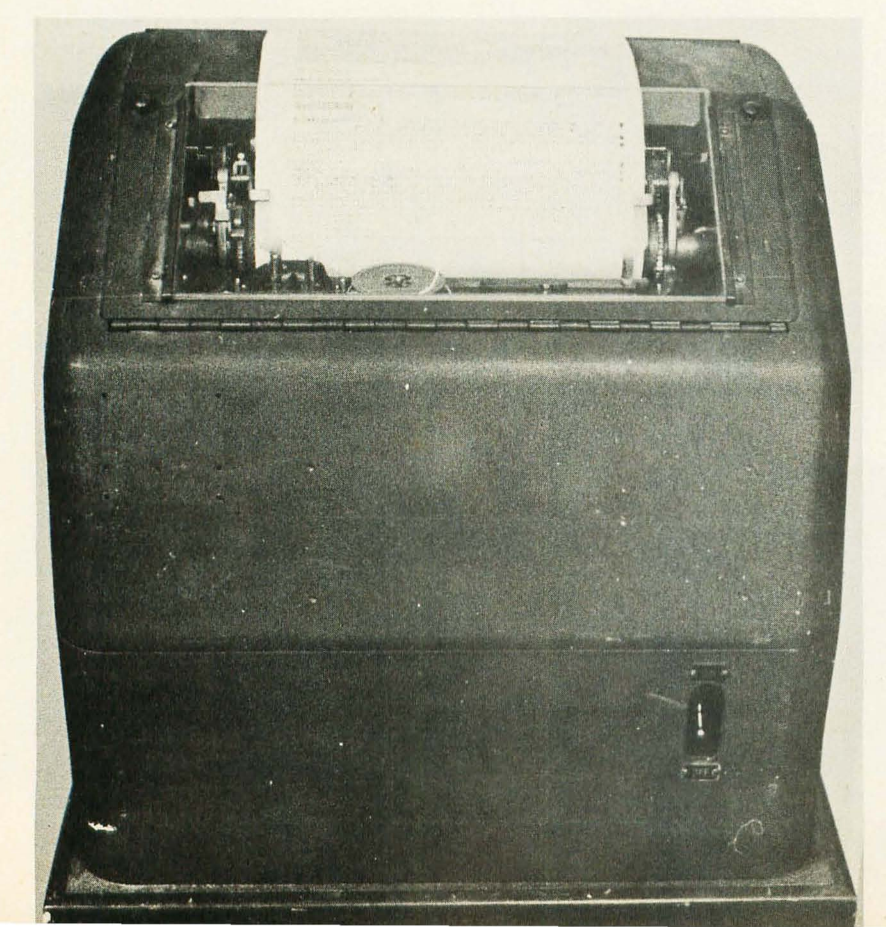
For more information, contact your local OSI dealer. Write OSI direct for our free OSI brochure. Or enclose $1.00 for the full OSI catalog.

Once again, OSI offers more and costs you less.



OHIO SCIENTIFIC INSTRUMENTS
Dept. B
11679 Hayden Street, Hiram, Ohio 44234

Photo 1: A Teletype Model 15 RO printer. This "receive only" model is shown with cover removed (above) and with the cover in place.



A Guide

Michael S McNatt
4658 E 57 St
Tulsa OK 74135

Are you wondering why you should consider yet another peripheral for your system when you may already have a nice, quiet video display and even a panel of Das Blinkenlights? Consider the following excellent quote by Robert H Cushman, one of the editors of *EDN* magazine, who has written several articles on designing with microprocessors:

> **Never Underestimate the Value of Hard Copy**
>
> *Working at the Teletype is a superior way to learn . . . software and, later, to put the microprocessor to use . . .*

	Bit Numbers 5 4 3 2 1	Lower (Letters) Case	Upper (Figures) Case: Communications (Standard)	Upper (Figures) Case: Bell System (TWX) Stock Market	Upper (Figures) Case: Weather Code
0	0 0 0 0 0	Blank	Blank	Blank	–
1	0 0 0 0 1	E	3	3	3
2	0 0 0 1 0	Line Feed	Line Feed	Line Feed	Line Feed
3	0 0 0 1 1	A	–	–	↑
4	0 0 1 0 0	Space	Space	Space	Space
5	0 0 1 0 1	S	' (Apos)	' (Apos)	Bell
6	0 0 1 1 0	I	8	8	8
7	0 0 1 1 1	U	7	7	7
8	0 1 0 0 0	Car Ret	Car Ret	Car Ret	Car Ret
9	0 1 0 0 1	D	$ WRU	$	↗
10	0 1 0 1 0	R	4	4	4
11	0 1 0 1 1	J	Bell	Bell	↙
12	0 1 1 0 0	N	,	7/8	○
13	0 1 1 0 1	F	!	1/4	→
14	0 1 1 1 0	C	:	1/8	⊕
15	0 1 1 1 1	K	(	1/2	←
16	1 0 0 0 0	T	5	5	5
17	1 0 0 0 1	Z	+	"	+
18	1 0 0 1 0	L	)	3/4	↖
19	1 0 0 1 1	W	2	2	2
20	1 0 1 0 0	H	#	#	↓
21	1 0 1 0 1	Y	6	6	6
22	1 0 1 1 0	P	0	0	0
23	1 0 1 1 1	Q	1	1	1
24	1 1 0 0 0	O	9	9	9
25	1 1 0 0 1	B	?	5/8	⊕
26	1 1 0 1 0	G	&	&	↘
27	1 1 0 1 1	Figures	Figures	Figures	Figures
28	1 1 1 0 0	M	.	.	.
29	1 1 1 0 1	X	/	/	/
30	1 1 1 1 0	V	=	3/8	⊙
31	1 1 1 1 1	Letters	Letters	Letters	Letters

Table 1: A listing of the various Baudot key code assignments. The weather code column, taken from the Reference Data for Radio Engineers, *fifth edition, pages 35 to 38, table 22, includes wind direction and cloud cover symbols. In the communications column, the letters "WRU" stand for "Who are you?"*

to Baudot Machines:

Part 1, Description of Available Devices

The hardcopy record provided by the Teletype printer proved unexpectedly useful. Sample programs and their execution results . . . could be reviewed many days later, often clearing up today's cause of confusion . . .

Hardcopy records are especially helpful during this learning process because when one attacks some new feature of programming a particular machine, he or she invariably begins by probing and experimenting, making many false tries. At that instant, all inconclusive attempts seem like a complete waste of time. But later, as you advance, copies of your first attempts enable you to go back and explain earlier mysteries. Psychologically this is most comforting, because it wipes away the frustations that invariably build up when neither you nor the computer appear to understand each other.

EDN tears off the 6 foot or so strip of Teletype paper that results from the typical bout with the computer, folds it accordionstyle, and files it. To make our reference complete we start each session by entering the day's date and the session's objective, and end it with a summary of what was, or wasn't, accomplished.

The compact hardcopy printout that you get is a much more useful debugging tool than the transient

Photos 1-5 courtesy Donald Clum. Photo 6 courtesy Robert W Cook.

displays of console lamps, or the temporary frames of similar video readouts. You can study this hardcopy at leisure, make notes on it, and then file it away for future reference.

All these practical virtues of performing program development through typewriter-like terminals have been known to computer types for two decades. These old timers are amazed to learn that it is taking the newcomers to the computer world so long to catch up.

["How Development Systems Can Speed Up the µP Design Process," *EDN*, April 20 1976, pages 63-72.]

Beware: Model 15s May be Hazardous to Your Health!

If you are considering a Teletype Model 15 or similar surplus page printer, be forewarned: Not only is the Model 15 obsolete (Teletype Corp no longer manufactures parts), but:

- *It's a rude, ugly mechanical monstrosity next to LED readouts, shiny microprocessor cabinetry, and solid-state keyboards.*
- *It's a veritable assault of clanking, grinding, and roaring among the silent flash of panel lights, video displays, and the barely audible hum of power supply transformers.*
- *Foul smelling oil and burning grease pour out of its bowels to drown the clean, woodsy smell of new programming books and operating manuals.*
- *It crawls along at a yawning, hair pulling, nail biting 60 to 100 words per minute, while a blazing processor hurries along wasting thousands of cycles between character outputs.*

But a Model 15 hardcopy page printer will cost between $35 and $100 depending on your resourcefulness, and luck.

Table 2: Physical characteristics of the four common teleprinter speeds for Baudot machines. In many cases, the same machine can be run at several different speeds simply by changing gears.

Words Per Minute	Data Rate	Characters Per Second	Bit Pulse Width	Stop Pulse Width	Operations Per Minute
60	45 bps	6	22 ms	31 ms	368
66	50 bps	6.6	20 ms	30 ms	400
75	57 bps	7.5	18 ms	25 ms	460
100	75 bps	10	13.5 ms	19 ms	600

What Kinds of Baudot Machines are Available?

Perhaps the best way to answer the question of what's around is to catalogue each machine briefly, starting with the Teletype Corp models and following with units by other manufacturers such as Kleinschmidt, Creed, etc. In these descriptions, I've included photographs of many, but hardly all, of the typical machines. Most of these units will be found with nameplates and number plates identifying the model and manufacturer (if you purchase a whole machine).

Model 11

The first machine introduced with the name Teletype, in 1921. Used a type wheel and printed on paper tape at 40 words per minute.

Teletype Model 12

Introduced in 1922, about 12,000 of these units were sold. Practically all Model 12s had been removed from service by the 1940's, which makes it a real antique. The unit consisted of a Baudot page printer and a keyboard, and sounded like a cement mixer. The electrical noise or hash generated by these machines would require extensive shielding and cables for use with a microprocessor. Give it to a museum if you find one. . .

Teletype Model 14

This series of Baudot paper tape readers and punches were manufactured beginning in 1925, with about 57,000 Model 14s of all types being sold. The Model 14 series consists of:

- Model 14TD, or "transmitter distributor" reads the paper tape and transmits serial data on a current loop. Speeds of 60, 66, 75 or 100 words per minute can be selected by installing the appropriate set of gears. Easily modified for parallel output. Price range from $20 to $150.
- Model 14 RO "Reperforator" or receiver, punches paper tape from serial data received from a current loop. In 1940 a typing reperforator was introduced which also prints on the tape, but the printing is several characters behind the punches. Speeds are 60, 66, and 75 words per minute. Price range from $10 to $50.

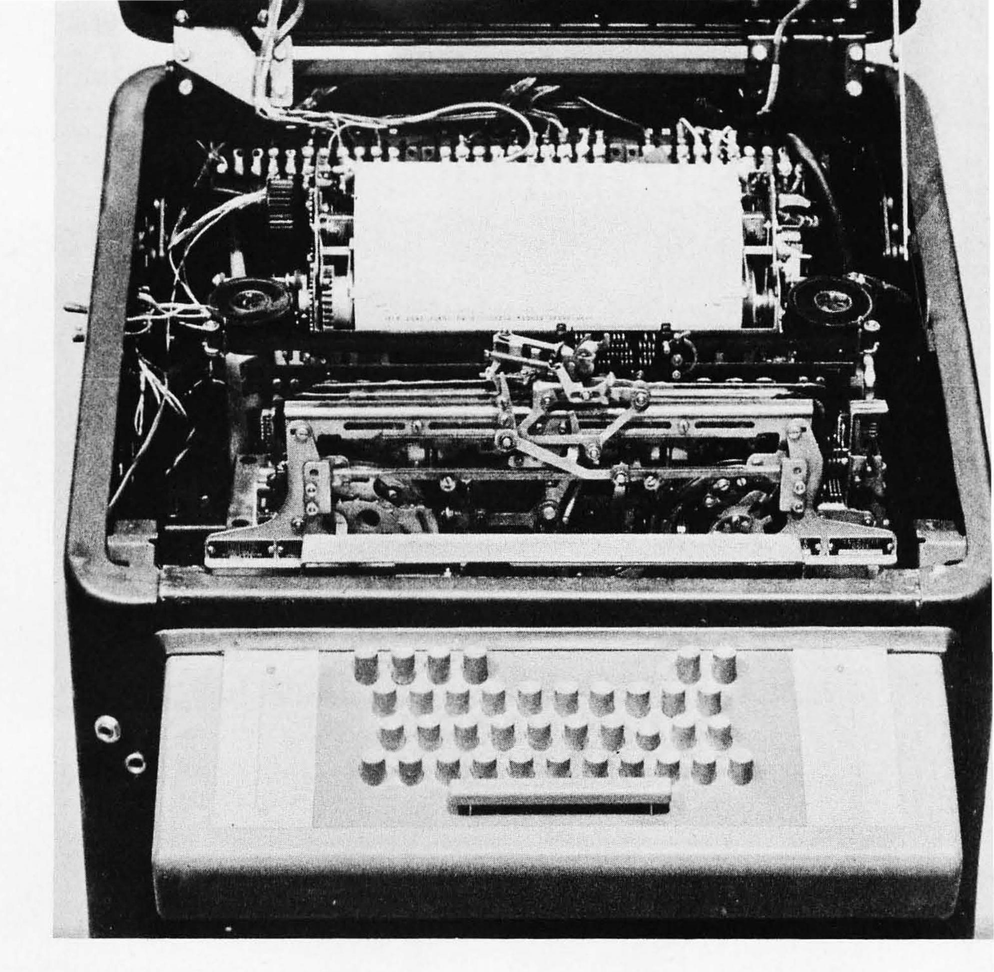


Photo 2: The author's Model 28 KSR Teletype machine. Note the much more compact mechanism (above) of this model compared to a similar view of the previous generation's Model 15 in photo 1. The Model 28 KSR with cover in place is shown below.

- Model 14 KSR: A Model 14 RO with a keyboard. Enables punching of paper tape off line. I've seen prices range from $15 to $75.

Teletype Model 15

This machine comes in two versions: a Model 15 KSR with page printer and keyboard and a Model 15 RO with page printer only (see photo 1). About 237,000 Model 15s were produced from 1930 to 1957, with Bell Systems receiving about 100,000 of these, the US government about 60,000, and Western Union, the wire services, newspapers, railroads, airlines and RCA Corp making up the balance. The page printer is usually set up for 60 words per minute, but occasionally 66 and 75 words per minute machines are encountered. The typical used price ranges from $30 to $200.

Teletype Model 19

A set of Baudot machines which includes a Model 15 page printer and keyboard, a paper tape perforator, a Model 14TD reader, and a heavy duty loop supply. On some Model 19s, the punch is mechanically linked to the keyboard. Thus, tape must be punched locally and cannot be punched via computer control. The punch magnet requires a 100 VDC source capable of supplying one amp. The tape reader is usually connected in series with the keyboard so that serial Baudot signals can be generated from either unit. The surplus price ranges from $125 to $250.

Teletype Model 20

This model was used by the wire services and was similar to the Model 15 except it used a six level code for an expanded character set. This is a rare machine and I have not seen it advertised.

Teletype Model 26

Another rare, light duty machine used by the wire services.



Photo 3: The "standard" Teletype so often seen in computer centers and personal computing contexts is the Model 33 ASR shown here with a modem option installed. This light duty machine is the war horse of the minicomputer world.



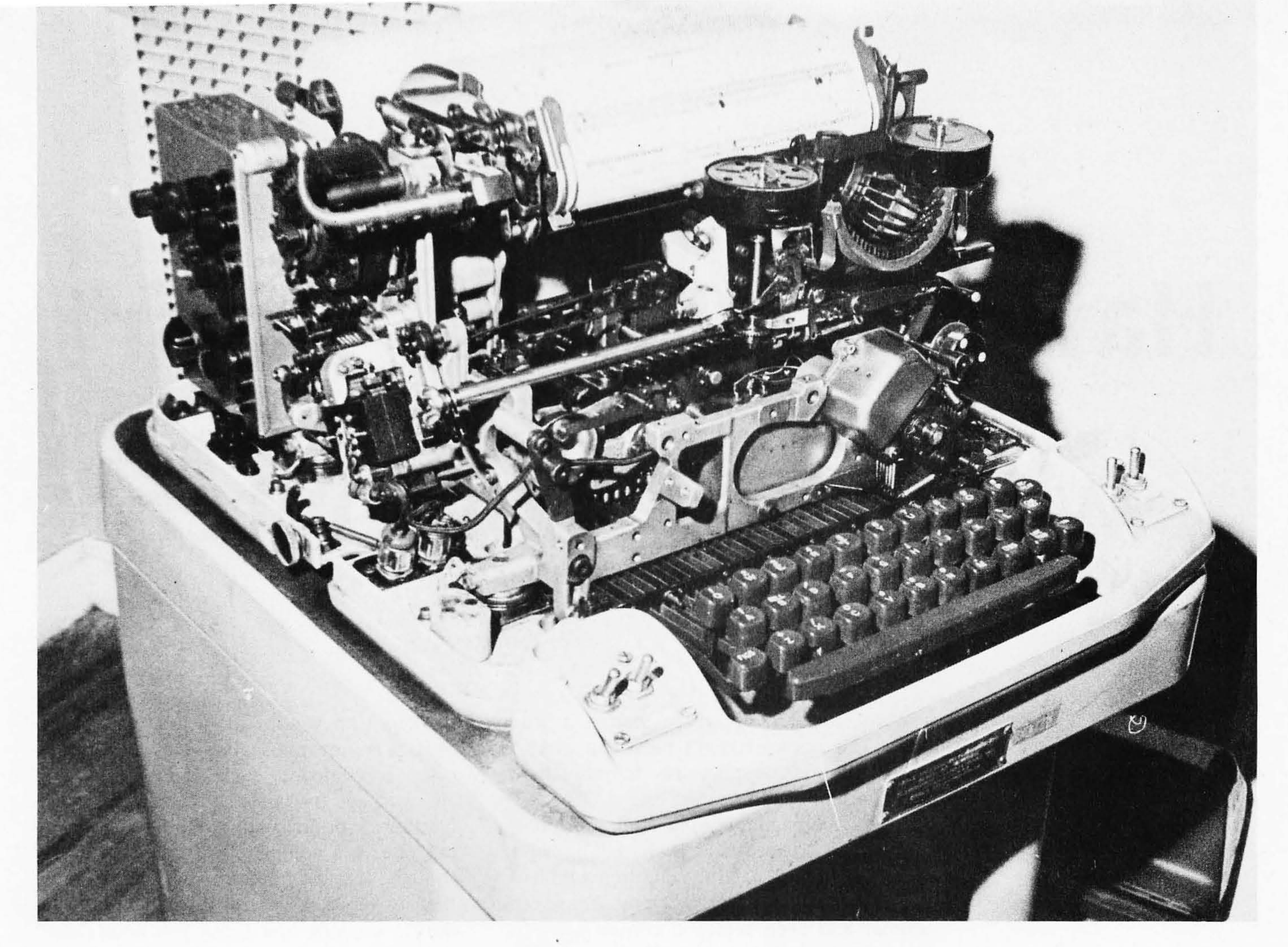


Photo 4: A view of a Kleinschmidt Model 150 KSR. This variant of the basic Teletype design (note its similarity to the Model 15) contains a current loop power supply.

Teletype Model 28

A general number for a series of Baudot machines, which includes ROs, KSRs, ASRs, TDs and reperforators. These are heavy duty machines like the Model 15, but generally operate at 100 words per minute. This is the most prevalent Teletype model used by the MARS people. Since production began in 1952, almost 300,000 Model 28s have been sold. This machine is still being produced so new replacement parts are plentiful. Photo 2 shows a Model 28 KSR.

There is an actual advantage of owning a used Model 28 as opposed to the popular Model 33 ASCII machines. This is because the Model 28 is a heavy duty machine made for 24 hours per day operation, year after year. However, the Model 33 is a light duty machine, which means it was designed for limited use: 3 to 4 hours per day, average. It will work for reasonably long periods of time but will require maintenance and adjustment more often than the Model 28 or any of the other heavy duty machines such as the Models 14, 15 and 19.

New prices from the Teletype Corp at last report were: Model 28 RO, $1750 to $2000; 28 KSR, $2000 to $2250; 28 ASR, $3300 to $3900; prices vary with options; delivery is from 6 to 12 months. None are in stock and each machine is built to order. Used prices range from $200 to $1500.

Teletype Model 29

Another rare machine, essentially a special Model 28 built for IBM.

Teletype Model 31

A rare teleprinter and keyboard combination, designed to print on 0.25 inch (0.64 cm) paper tape. This was a small piece of military aircraft equipment (nomenclature TT-30/AGA-1) which was designed shortly after World War II. It measured 9 by 14 by 12 inches (23 cm by 36 cm by 31 cm) and weighed 23 pounds (10.4 kg). Although assigned the Teletype number 31, there is little indication that there was ever any civilian production.

Teletype Model 32

Looks just like the ASCII Model 33 except it has a three row Baudot keyboard. This is a light duty, 100 words per minute Baudot machine still in commercial use. Available as RO, KSR and ASR. Used price

Continued on page 154

Having a "Private Affair" with Your Computer

Steve Ciarcia
POB 582
Glastonbury CT 06033

Then I made a mistake! I told someone that I had a computer!

What you are about to read is a true story. The names haven't been changed to protect the innocent because, frankly, there aren't any innocent parties.

It was the beginning of a cool autumn in Connecticut. The leaves were falling with increasing rapidity and the children who had played noisily in front of my house all summer had been removed conveniently to schools well out of earshot. There was just a slight rustle in the leaves and now was the time to contemplate the future without the disturbances of the present.

Many technical individuals are probably like me – the quiet and solitude stimulate the mind. Through your consciousness race thoughts of all kinds of inventions which haven't even been thought of yet. But you resign yourself to living quietly with a few modest pleasures which complement this attitude.

I consider a home computer system to be within my definition of modest. It is just a small 8080 system with 10 K and supported BASIC. I didn't intend it for any particular purpose, just relaxation. I assembled it and placed it in my shop with my other modest pleasures and plinked away on the keyboard occasionally. It was more the challenge of getting the computer up and running than writing exotic programs to keep an inventory of the kitchen cupboards, or calculate the surface area of the family car. I put together a CRT and cassette mass storage system and was content just generating peripherals. Then I made a mistake! I told someone that I had a computer!

While at lunch one day and after a few martinis, I let it slip to a group of fellow workers I knew a lot about microprocessors. This was not unusual – the conversation often gravitated to such topics because we all worked for a large computer manufacturer. Working with computers and owning one is a completely different situation though. It's just ho-hum when cranking out the answer to a 40 by 40 matrix in 10 ms during working hours, but this same person goes bananas when getting the correct answer to 2 times 2 in BASIC on a home computer. It's absolutely crazy – but then again, I'm an engineer, not a psychiatrist.

One member of the group, Ralph, continued to quiz me about what I had said. He knew from past experience that when it came to hardware, I wasn't an idle conversationalist. Since he asked and I was in an exceptionally happy and unguarded mood after a good lunch with pleasant company, I told him that I knew about microprocessors because I had one. Well, the cat was out of the bag.

"Waiter! Bring my friend here another martini!" he shouted as his interest began to heighten.

I'm sure everyone knows or has worked with a character like Ralph. He's a mild mannered suburban family man whose goal is to own the world. It's not an idle fantasy, but rather a calculated approach to life – all of it dollar signs. The day to day moods of people like this are in direct proportion to the Dow Jones daily average. When the Dow is down, don't mention anything controversial, lest it be interpreted as a personal affront challenging his integrity. On a good market day, Ralph is the wise philosopher who has a kind expression of optimism for

everyone with whom he comes in contact. It is during these latter periods that everyone must be wary. His generosity is overwhelming even though the drinks he orders end up on someone else's check, and his money-making schemes get more and more prolific.

It's only the little bald spot on the top of his head which really seems to change. I guess it's a simple case of not getting older — just getting better. If there is a way of turning a profit on that, Ralph will find it.

Ralph held back the torrent of questions until the third martini was half consumed. And, preferring not to be too obvious, he just asked leading questions calculated to make me talk. I was brilliantly coy, even though a bit giddy and was quite able to fend off the barrage of questions until he said, "I'm sure that even though you may have a microprocessor — and I'm sure you only mean the microprocessor chip — that it would be virtually impossible for even an experienced engineer like yourself to put together some sort of computer that could conceivably be worth anything."

"What do you mean, couldn't build it!" I said, defending myself without realizing what I was about to say. "I have already built it, brother! I have a running 10 K system with a CRT, keyboard, and cassette storage system. I'm running BASIC now and have a few games like blackjack and . . . and . . ."

Oh, Oh! I did it, and I knew it . . . Ralph's ear twitched just like someone pulled the handle on a one-armed bandit and his eyes began to spin. When they stopped, it was like looking into two dollar signs and Ralph had hit the jackpot. "And . . . if you don't believe me, come over tonight and see it!"

Why I ever said that I'll never know. I could have kicked myself and had even considered ordering a half dozen martinis hoping to be rushed to the hospital with acute alcohol poisoning just to find a good excuse not to keep our engagement. I was a blabbermouth and should face the music, so I shook off the suicidal tendencies and went back to the office. All afternoon while I was consumed with a dull headache, I couldn't help but notice that Ralph's little bald spot was a brighter pink than normal — a rare occurrence, usually only after a big killing in the market.

By that evening, I had regained my composure and decided to make the best of it. Maybe if I just load a silly game written in BASIC and entertain Ralph for a while, I could convince him that this computer is a toy. You know, a toy like Odyssey or Pong.



Just as I had finished loading the game, Ralph came downstairs into my shop carrying a briefcase stuffed beyond normal capacity, an armload of papers, pamphlets and other notebooks. I didn't bother to ask what all the junk was for, I just greeted him and welcomed him to the inner sanctum. Ralph was very familiar with computer rooms and industrial facilities, but he could not cease to be amazed at the quantity of equipment that I had amassed in my basement.

"Well, with all this junk, you must really have a computer," he pronounced as a subdued compliment. "Do you have it working? Can I see the instruction set for the BASIC? How many lines of BASIC can be put in 10 K?"

"Hold it, Ralph! One thing at a time. Here's a copy of the instruction set. Why don't you sit down here at the keyboard and play blackjack while I go upstairs and make us some coffee." Going and getting a cocktail would probably have been a more hostly gesture, but after the luncheon, the thought of another martini was painful, so I resigned myself to coffee.

I'm basically a perfectionist and as with the computer, so goes the coffee — or is it the other way? Well, I mixed scoops of Jamaican Blue Mountain with Mocha and a touch of Deep Java and ground the coffee. All the while, I could hear the keyboard keys being pounded madly and a little verbal noise, somewhat like a kid talking to himself while doing a difficult math homework problem. It took about a half hour to prepare this exquisitely brewed concoction as well as bake a special tea biscuit (from an old family recipe, of course) which I always like to serve guests.

I couldn't help but notice that Ralph's little bald spot was a brighter pink than normal — a rare occurrence, usually only after a big killing in the market.

Out go games, relaxation, pleasure – in come the stock market, aggression, ulcers.

I have never been to a horse track, dog track, betting parlor, pool parlor, Las Vegas, church bingo game, or even bought a raffle ticket.

When I reached the area of my work bench with the tray of biscuits and coffee, I glanced at the CRT. My blackjack program was no longer on the screen, but rather a simple little program which took daily stock quotes and computed weekly and monthly averages.

"I was only gone a half hour," I exclaimed.

"Oh? Yeah, I learned BASIC when I was in Australia building an oil refinery one time. It was the same time that I got into the foreign stock markets. There's not much else to do down there, you know." He had a broad smile on his face which was an obvious result of having mastered the computer so easily, and he took the coffee and biscuits. He pounded the keyboard, chomped biscuits down in one gulp, slurped noisily, and shuffled papers madly while his head continually bobbed up and down looking at the CRT.

"I think I can really do something with this system. I want a little time to study the instructions some more, then I'd like to come over and discuss some businesss applications with you." This seemed to be a reasonable request and I agreed, but before he could leave, we had to make a copy of his program on cassette tape.

"Gee, I only have this one cassette which has the blackjack on it, but I suppose we can use it. I can always load it by hand. It only took a half an hour." I sighed and inserted my cassette and hit the record button. Out go games, relaxation, pleasure – in come the stock market, aggression, ulcers. Oh well, it's only a cassette tape and a lousy computer game.

"Wow, Ralph, it's almost 12:30. It's been a long night. Glad you could come over. Study up and maybe we'll get to do it again soon." I nearly choked on the last comment. Actually, I'm really not that bitter and warped. I really like Ralph, and he means well. But to be on the safe side, I'll buy a half-dozen cassettes tomorrow.

The saga continues . . .

Two nights later, Ralph came over again with the same enthusiasm and more notes and papers. Ralph had a funny look in his eyes and his bald spot was unusually pink, but I didn't pursue it. It was probably coming in from the cool night air to the warm house. The greetings and salutations were hardly over when Ralph had sat down in front of the computer and announced that he was ready to go. I asked if he wanted me to load his stock program which we had put on the cassette from the previous session, but he protested that was child's play and he was really ready to program now. In fact, he had written a lot of it already and opened one of his notebooks to show me five solid pages of BASIC programming.

"And that's just to enter the player statistics and post positions," he exclaimed, delighted that he had accomplished so much in so little time and indicating as subtly as possible that this was just the beginning.

"Statistics? Post position? For what?" I asked.

"For jai-alai, of course!" he exclaimed and I immediately knew that the odd look and pink bald spot had nothing to do with the cool night air. "We enter the players' statistics, their post positions, and all their previous performance records, the computer does a little mathematical hocus-pocus and plays the game and predicts the winners." He could hardly hold still as he was describing this mind-boggling scheme to me. "And when we add the betting section to the program, it will tell exactly how much to bet on the trifectas, perfectas, and quinellas. And . . ."

"Hold it! You mean you are going to write a game program like my blackjack program?" I asked. A game program was only natural as far as I was concerned. I was a true nongambler and never bet on anything but a sure thing. Even then, it wasn't really a bet because the things I would place a wager on entailed no risk and I'd be an idiot not taking a chance. Such things were betting that my salary this week was the same as last week, or with the gas station attendant that the car needed a quart of oil (I checked it before going to the gas station), or that it would snow in Connecticut this winter. I have never been to a horse track, dog track, betting parlor, pool parlor, Las Vegas, church bingo game, or even bought a raffle ticket. Sometimes I think I lead a protected life.

"Hell no! I mean a program to play and bet real money at the jai-alai fronton here in Hartford!" he exclaimed almost jumping off his seat. There was a point at which I wondered whether I should fear for my life and considered calling my dog, but affectionate Scottish Terriers don't seem to demand the same respect as Doberman Pincers.

"This is only a 10 K system. We don't have enough memory to do a program like that," I rebutted, hoping to clip this crazy proposal in the bud.

"Oh, don't worry about that. We'll add some more memory. I figure it will take about 24 K and of course, we'll eventually need some sort of hard copy device . . ."

"OK..OK.. go program something while I think about it," I said in a resigned sigh. I

MERLIN

The Intelligent Video Interface

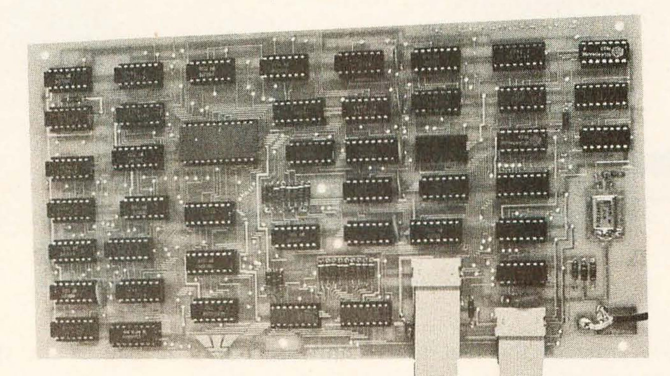
1v p-p Composite Video for standard monitors. Syncs and video signal are available separately on the test and expansion connector.

MEI 2K x 8 Mask ROM
Extended Monitor & Editor functions plus graphics and cassette subroutines (*$34.95*).

Serial Expansion Kit provides 3 parallel input ports and 2 parallel output ports via KYBD & Serial I/O Connector (*MSEK — $45*).

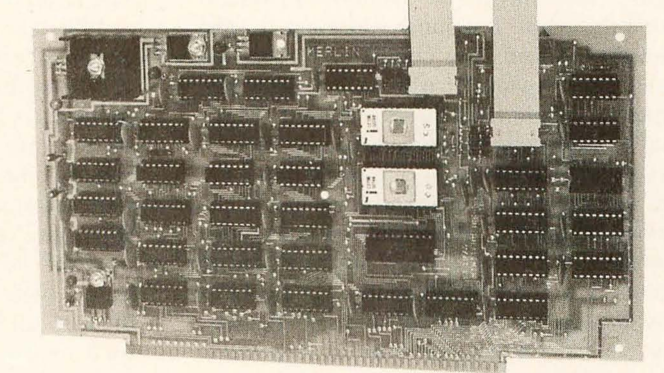
Graphic bit mapping Circuitry — 160 H x 100V resolution

8-Bit Programmable Mode Register
Program MERLIN for ASCII, graphics and special display formats.



Super Dense Graphics Kit
increases MERLIN's resolution to 320H x 200V. This board plugs directly into MERLIN at the test and expansion socket (*M320 — $39*).

Software Tarbell Compatible Cassette Interface
Interfaces via keyboard cable. Software is available in the MEI ROM (*MCAS — $29*).



200 Page, 3-ring User Manual tells how MERLIN works and how to use it ($10).

MBI 2K x 8 Mask ROM
System Monitor and Editor plus scrolling and I/O drivers.

S-100 Bus Compatible

KYBD & Serial I/O Port has 8 bit parallel input for keyboard, + 5v, -12v, and data and clocks for serial expansion.

MBI ROM/RAM — $39.95
(Eliminates need for expensive **Monitor Boards.)**

ROM Character Generator for ASCII Display mode 40 characters by 20 lines

16 Bit Scroll Register DMA Circuitry —
Display any portion of your system memory.

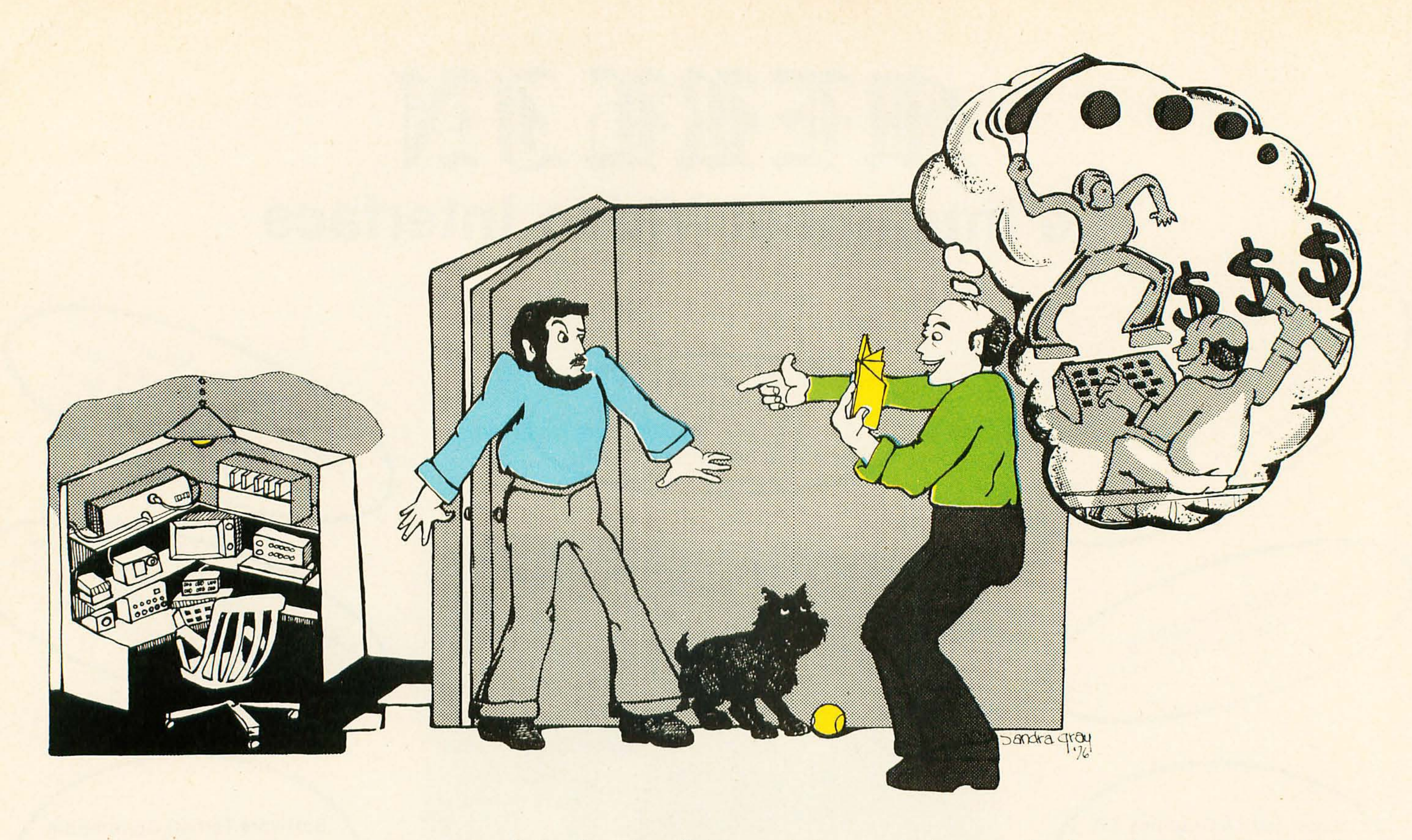
The MERLIN Kit with manual is $269.
MERLIN assembled & tested is $349.
Order yours today.
MC and BAC accepted

If you are still not convinced that MERLIN is the best buy in Video Interfaces/Monitors see one at your local computer store or write for more info.



MiniTerm Associates, inc.

Box 268, Bedford, MA 01730 (617) 648-1200



took note of the fact that Ralph used the term "we" when he mentioned system expansion, but I seriously doubted it. The thought of using my computer for betting was crazy. Obviously, if it were a good idea, someone would have already done it. That's the American Way. It sure seemed like a risky idea. I had visions of putting down my life savings at the betting window just as the computer takes a power hit and goes down complete with my money and my sanity. Ralph is a good programmer, but he can't possibly do the entire job himself. Besides, my system is much too small to accomplish what he's attempting. Maybe I should let him plink away at the keyboard until he realizes that he can't do it and then I can have my system back and relax while I play Kingdom or something.

I had visions of putting down my life savings at the betting window just as the computer takes a power hit and goes down complete with my money and my sanity.

"Fantastic," yelled Ralph. "I've got the player entry and post position sections already working! I can't believe how easy it is to program this computer."

The little man was amazing. All day he sits in his office, shuffling papers and occasionally discussing business with customers, but I had never experienced this completely "off-the-wall" emotionalism.

"This is terrific! I can see that we'll have this program really going as soon as I get rid of a few bugs," he continued.

Bugs? Was that bugs he said? Salvation is what those bugs were. With a program bug, it might never work. I might yet get to use my system again. "Gee, Ralph. That's too bad about the bugs. I guess you won't be able to do it after all."

"I'll take care of the program bugs, but you had better check some of the hardware. I've been dumping the program to cassette about every 30 minutes so that I could keep an accurate revision record or in case of a power failure and I noticed that your cassette interface drops a bit once in a while. When you've got a lot of money on the line, we can't afford dropping bits," he interjected with an authoritative resolve. "I've used up all but 300 bytes of the 10 K and we'll need some more space if you want to do this right. Didn't you say that you wanted to expand the system to put larger game programs in it?"

He was correct. I did eventually want to expand the system with more memory, but I would have been happy just playing blackjack by myself right then. I couldn't even do that. The six cassettes I had just bought were conveniently written with various versions of jai-alai. Ralph was correct — my $39 El Cheapo cassette recorder probably did drop a bit or two, but when you're playing blackjack, who cares? Still, I couldn't have someone criticize my hardware. I was a good hardware designer and didn't want anyone to think otherwise; especially someone I worked with.

Ralph gathered up his paper work and as he left, he tossed me a jai-alai program to study so I could understand the betting terms and be able to use the program

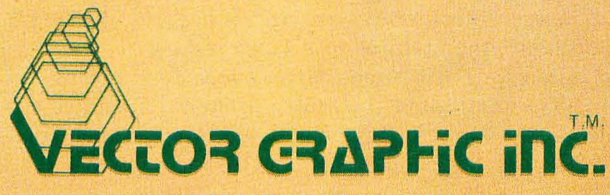
Now. THE PERFECT MICROCOMPUTER VECTOR 1



It's a two touch affair

VECTOR 1 is based on the 8080A Microprocessor and S-100 Bus -

EASY TO ASSEMBLE, EASY TO ACCESSORIZE, EASY TO BUY AND VERY EASY TO OPERATE.



717 LAKEFIELD ROAD, SUITE F
WESTLAKE VILLAGE, CA 91361 • (805) 497-0733

I didn't want any more cassette storage criticism, but telling Ralph about my depleting bank account would have fallen on deaf ears.

Working with Walt is a little nerve-wracking because in computer terms, he is a LIFO, or "last in, first out" interrupt processor.

eventually. I didn't say anything except goodnight and come again sometime. Sometime would be soon he assured me and I felt a sudden pain in the center of my forehead. I never used to get headaches. Maybe I needed a vacation . . . .

The next four evenings I diligently worked at improving the system and increasing the memory. The leaves were piling up around the front door and I knew I should be raking them or putting antifreeze in the cars, or something. I hadn't picked the ripe tomatoes from the garden before last night's freeze, but now I don't have to worry about them anymore. I saw my wife this evening in the hall on my way down to the workshop. She looked pleasant and gave me a half-hearted smile. It was her subtle way of saying, "I remember you. Wasn't I married to you some time ago?" I just didn't have the time to explain because Ralph was on his way over and I wanted to show him all the improvements, so I patted the dog and descended the stairs into the workshop.

I looked around at all the equipment and some of the unfinished projects on my desk. For a couple of months, I had been developing an analog interface to be attached to the computer to allow joystick inputs to liven up future game programs I was contemplating. It was all done, just awaiting checkout and debugging, but I needed the computer to check it dynamically, and with all the stuff with Ralph lately, I hadn't been able to even get close to it. I still had an older 8008 system with 4 K of memory which I could use, but it was located right next to the 8080 system and with Ralph sitting there, I couldn't even get close to it. I resolved that I would sit at my desk and recheck the wiring again.

"OK, are we ready to go?" asked Ralph as he came into view while descending the stairs. "I hope you don't mind — I invited Walt along to help me debug the software. He's a genius when it comes to programming."

Ralph was quite correct; Walt was indeed a software genius. Walt is one of those truly helpful people who continually goes out of his way to accommodate everyone. Working with Walt is a little nerve-wracking because in computer terms, he is a LIFO, or "last in, first out" interrupt processor. The poor guy will be diligently working on something important, and if someone asks for his help, he drops everything, solves the other person's problem, then returns to his own task. Some interesting situations have evolved when he's interrupted by five different people each asking for his help. I suppose that's the price of genius. I just chuckled to myself while thinking what kind of line Ralph must have laid on him to get him to work on this crazy scheme.

"So, did you get a chance to fix the bit problem?" Ralph asked.

"Yes," I answered. "It was nothing that a better tape recorder wouldn't cure," I replied, still wondering whether replacing my low bid recorder was really necessary. The replacement was a $149 Sony tape recorder — the best one I could find. I didn't want any more cassette storage criticism, but telling Ralph about my depleting bank account would have fallen on deaf ears. I also added another 8 K of memory and beefed up the power supply. That was only about $300.

"Great! We'll really be able to make progress now," he chanted. "We'll give you a cut of the winnings to help cover the hardware costs," he continued. "Hey Walt, sit here at the desk under the light and see if you can find the bug."

Ralph loaded the last program version and spread his papers out all over my bench. Papers hung over the top of my oscilloscope, propped against my parts cabinets and thoroughly covered my work area. Walt asked if it was OK to move the analog interface board which was on the desk, so that he could spread out some papers and not have to worry about accidentally leaning on something. "Sure Walt, let me move everything for you," I said as I cleared off the desk. "Can I get anyone a drink or something?"

"I'd like a beer if you have one," said Walt.

"I'll have a gin and tonic and if you have any of those biscuits left, that would be tremendous," said Ralph.

I slowly walked upstairs and stopped on the landing to pat my dog. Scottish Terriers are like people. They can communicate with their eyes and he was giving me that "You don't play with me anymore" look, and dropped a tennis ball by my hand hoping I would have a favorable reaction to his request. I used to work at my bench and occasionally kick a ball around the cellar for him. Now we both knew that we couldn't do that any more and we sat watching the two men intently discussing the program and inspecting the CRT display. I went upstairs to prepare refreshments and invited the dog to follow, hoping to placate him with a fresh baked biscuit.

My wife walked into the kitchen and being the understanding soul that she is, asked me if I could possibly light a fire in the fireplace and use the den for whatever it was that I did downstairs.

"That's it! That's how I can do it!" I yelled. If I was Polish, I would have danced a polka, but instead I grabbed the tennis ball and threw it down the hall and the dog made a mad dash after it.

"I can't use the den, but they can! All I have to do is put a remote terminal in the den." It was an extravagant idea, but I was desperate. I would move my friends to another section of the house, out of my shop and out of my dog's playground.

I hadn't worked out the details yet, but that's what I was going to do. I had contemplated moving the computer, but like any hobbyist's, it was spread out all over three shelves with patch cords and was impossible to move. The details of the design flashed through my head and I hurriedly looked for a pad to record my thoughts. Twisted pair shielded wire, asynchronous interface, coaxial cable for the monitor, perf board, line drivers — Hold It! — one thing at a time. The design of the remote terminal was fairly simple, but would take a little time to build. I'd do that later. Right now, it was important to see if I had enough wire to go from the computer to the den and I made a mad dash down into the shop and started rummaging through my storage bins until I found a 200 foot roll of coax and a big tangled mess of what appeared to be twisted pair shielded wire. Ralph and Walt didn't seem to be bothered by all the commotion behind them until I pulled down half the bookcase while yanking on the tangled mass of wire.

I grabbed some tools and dragged the wire outside through the garage. Ralph and Walt had kicked the books out of the way and returned to programming by the time I was out on the front lawn. It was cold outside, but that was the only place with enough room to spread out the mess. I'm sure the neighbors thought it a bit strange to see a grown man out on the front lawn at nine o'clock in the evening with a flashlight doing what appeared to be playing in the leaves. My dog came out to investigate and with all his jumping around and barking, it really must have looked pretty bad, but I was determined.

It took quite a while, but finally I untangled the mess and had two twisted pair and one coax line stretching down the street past the neighbors' home. I decided to put in an extra twisted pair line in case I had other friends and had to go to time-sharing. I placed a ladder up against the side of the house under the den window, and with the wires between my teeth and an electric drill in one hand, attempted to scale the ladder. I swore that the next house I bought would be a ranch instead of a colonial or at least I'd put the den on the first floor, but for right now, I just didn't look down and concentrated on the "do or die" situation at hand.

I started the electric drill, placed it against the house and pushed! I drilled through the storm window, the inside window and half the window sill. It wasn't necessary to tell my wife that someone was outside the window, because I heard her



scream. Apparently, she was watching television in the den and the window I had just drilled through was behind the TV. It hit me all of a sudden what her reaction would be — it was something I had suggested she do if she were ever home alone or needed protection. She had probably rushed off for the bedroom closet to get the shotgun we kept there. Well, if you think firemen come down ladders fast, you should have seen me.

It took a while to calm her down, but she said that after having been married to me so long, she should have expected it. As a gesture of peace, I agreed to put off wiring the house until daylight hours and decided instead to design the asynchronous interface. Ralph and Walt had finished for the evening

and were apparently unaware of the preceding "fiasco" when Ralph said, "I hope your evening was as productive as ours. Walt figured out the bugs but it will take a few more sessions to incorporate the revisions. That's quite a system you've got there. You must really enjoy it."

I didn't get mad, I just looked tired. Even the dog decided he couldn't take anymore and went to lie down in front of the fire. "Well Ralph, it will be nice to see the program work so you can spend all your time at the jai-alai fronton."

I smiled from ear to ear and thought of the coup I was about to perpetrate.

"I'm looking forward to that too. But we still have a lot of work. Day after tomorrow evening is a good night for me. My wife may not be home, so I might bring the kids. Do you like kids?" he asked.

"Kids are fine. It's their paren. . .," I started as he interrupted.

"Great. Walt and I will see you then," was his final pronouncement as he stepped out of the door; but on the way to the car, he turned long enough to yell, "Hey, someone left this big mess on your front lawn, but if you pile enough leaves over it, nobody will know."

"Thanks, Ralph." I just closed the door, closed my eyes, and leaned against the wall. I judged that I didn't have much time before they were coming again, so I had better get to work on the communication link to the den.

During the next two days, I worked at a feverish pace, stringing wire and soldering components. I stuffed the three wires through the hole in the den window, ran the wires down the side of the house to the first floor level, and then drilled a hole through the outside wall into the garage. I had to go through the back and out the front of a garden tool cabinet in the garage, but my wife wouldn't notice that she couldn't open the door any more until spring – I'd just use my skill saw and make a new opening for her.

During the next two days, I worked at a feverish pace, stringing wire and soldering components.

Stringing the wire through the garage was easy. I just hung it across the ceiling and looped it through the garage door springs for support. It was only when I started chopping a hole through the concrete wall into the basement that I got upset. Some idiot put a layer of sheet metal in the center of the concrete and the drill bits wouldn't even penetrate it; but this turned out to be no match for a cold chisel and a five pound hammer. The hole did turn out to be a bit larger than I intended, but I always wanted to install one of those little doors where a dog lets himself in and out.

I quickly attached the twisted pair to the line receiver in the shop. The unit downstairs consisted of the receiver section of a UART (universal asynchronous receiver/transmitter) while the box in the den contained the transmitter and line driver. A keyboard is plugged in upstairs, and when a key is pressed, the ASCII keyboard data is serialized and sent to the receivers downstairs where it is reconverted to parallel for the computer. I chalked up the $40 it cost to build this interface to experience for the sake of family harmony. The $175 for another keyboard and another TV monitor was a bit harder to take.

Ralph, Walt and whoever else was coming over would be there soon. I decided on a lark to try to type in the blackjack program before they came and busily attached the keyboard. When I was through, I took the tape containing their oldest revision of jai alai and recorded my blackjack back on it . . . just in case. When I was done, I loaded their latest program version and ran up to the den to check that all was working correctly. The screen was brightly lit with the BASIC operating system indicating that the video link worked properly. Data entry on the keyboard also proved successful and I felt like a new man. I smiled from ear to ear and thought of the coup I was about to perpetrate. I made myself a Tangueray martini, sat by the fireplace and awaited the arrival of my honored guests.

It wasn't long before Ralph and his two kids were at the door with Walt closely behind. I hadn't even closed the door before the dog and the children were off chasing each other through the house. "Are all kids like this?" I asked.

"Only happy ones," answered Ralph. "So do you want to come downstairs and see how this works?" he inquired.

My opportunity was there, but with all the noise I couldn't think and just said, "Tonight we are going to put you in the den. I put a monitor and a keyboard up there and you guys work upstairs while I work in the shop!" The words were finally out. I had stated my declaration of independence and I had demanded my workshop back.

"Sure. That sounds OK with us," stated Ralph as he and Walt spread out their papers all over the den. "It sure is nice up here. How do you keep a den looking so nice? Anyone with kids can't have furniture like this," continued Ralph as he threw his briefcase on the velour covered Louis XVI chair next to the fireplace.

I looked at him and stated, "I don't have any children, remember?" Just then, a tennis ball whizzed by with two kids and a dog in hot pursuit. Usually the dog wasn't even allowed on the oriental rug, but who was I

The Small Computer

Twenty-five years ago a computer as powerful as the new Processor Technology Sol-20 priced out at a cool million.

Now for only $995 in kit form or $1495 fully assembled and tested you can have your own small computer with perhaps even more power. It comes in a package about the size of a typewriter. And there's nothing like it on the market today. Not from IBM, Burroughs, DEC, HP or anybody else!

It fills a new role

If you're an engineer, scientist or businessman, the Sol-20 can help you solve many or all of your design problems, help you quantify research, and handle the books too. For not much more than the price of a good calculator, you can have high level computer power.

Use it in the office, lab, plant or home

Sol-20 is a smart terminal for distributed processing. Sol-20 is a stand alone computer for data collection, handling and analysis. Sol-20 is a text editor. In fact, Sol-20 is the key element of a full fledged computer system including hardware, software and peripheral gear. It's a computer system with a keyboard, extra memory, I/O interfaces, factory backup, service notes, users group.

It's a computer you can take home after hours to play or create sophisticated games, do your personal books and taxes, and a whole host of other tasks.

Those of you who are familiar with small computers will recognize what an advance the Sol-20 is.

Sol-20 offers all these features as standard:

8080 microprocessor — 1024 character video display circuitry — control PROM memory — 1024 words of static low-power RAM — 1024 words of preprogrammed PROM — built-in cassette interface capable of controlling two recorders at 1200 bits per second — both parallel and serial standardized interface connectors — a complete power supply including ultra quiet fan — a beautiful case with solid walnut sides — software which includes a preprogrammed PROM personality module and a data cassette with BASIC-5 language plus two sophisticated computer video games — the ability to work with all S-100 bus products.

Full expansion capability

Tailor the Sol-20 system to your applications with our complete line of peripheral products. These include the video monitor, audio cassette and digital tape systems, dual floppy disc system, expansion memories, and interfaces.

Write for our new 22 page catalog.
Get all the details.

Processor Technology, Box B, 6200 Hollis St., Emeryville, CA 94608. (415) 652-8080.



THE BASIC SOFT



The "HOW-TO-DO IT" Books for the "DO-IT" Person

* * Available NOW! ! ! at most computer stores * *

ARIZONA
Byte Shop of Phoenix-West
12654 N. 28th Drive
Phoenix, AZ 80529
(602) 942-7300
Jeff McKeever

Byte Shop of Phoenix
813 N. Scottsdale Rd.
Tempe, AZ 85281
(602) 894-1129
Alan Hald

Desert Data Computer Store
P.O. Box 1334
Tucson, AZ 85702
Gary Miller & Bud Ward

CALIFORNIA
Byte Shop of Lawndale
16508 Hawthorne Blvd.
Lawndale, CA 90260
(213) 371-2421
Dick Moule

The Computer Store
820 Broadway
Santa Monica, CA 90401
(213) 451-0713
Dick Heiser

A-VID Electronics
1655 E. 28th St.
Long Beach, CA 90806
(213) 426-5526
Reynolds Johnson

Byte Shop of Pasadena
496 S. Lake Ave.
Pasadena, CA 91101
(213) 684-3311
Govinda Singh

Byte Shop/Thousand Oaks
2707 Thousand Oaks Blvd.
Thousand Oaks, CA 91360
(805) 497-9595
Joe Kappl

People's Computer Shop
13452 Ventura Blvd.
Sherman Oaks, CA 91423
(213) 789-7514
W. K. Ling

Byte Shop of San Diego
8250 Vickers - H
San Diego, CA 92111
(714) 565-8008
John Perry

Byte Shop of Westminster
14300 Beach Blvd.
Westminster, CA 92683
(714) 894-9131
Marty Rezmer

The Computer Mart
625 W. Katella #10
Orange, CA 92667
(714) 633-1222
Allan Tarsky

Byte Shop of Santa Barbara
4 West Mission
Santa Barbara, CA 93101
(805) 966-2557
Scott Daley

Byte Shop of the San Fernando Valley
18424 Ventura Blvd.
Tarzana, CA 93156
(213) 343-3919
Tom Mardy

Byte Shop of Fresno
4131 E. McKinley Ave.
Fresno, CA 93703
Mike Sannes

Byte Shop of Mt. View
1063 W. El Camino
Mt. View, CA 94040
(415) 969-5464
Boyd Wilson

Byte Shops, Inc.
1261 Birchwood Dr.
Sunnyvale, CA 94086
(408) 734-9000
Bryan Kerr

Byte Shop of Palo Alto
2233 El Camino Real
Palo Alto, CA 94306
(415) 327-8080
Bob Moody

Byte Shop of San Mateo
1200 W. Hillsdale Blvd.
San Mateo, CA 94403
(415) 341-4200
Chet Harris

Byte Shop of Hayward
1122 "B" St.
Hayward, CA 94541
(415) 537-BYTE
Mike Lipschutz

Byte Shop of Diablo Valley
2989 North Main St.
Walnut Creek, CA 94596
(415) 933-6252
Dan Soldalh

Byte Shop of Berkeley
1514 University Ave.
Berkeley, CA 94703
(415) 845-6366
Pete Hollenbeck

Byte Shop of San Rafael
509-B Francisco Blvd.
San Rafael, CA 94901
(415) 457-9311
Don Anderson

Byte Shop of Santa Clara
3400 El Camino
Santa Clara, CA 95051
(408) 249-4221
Todd Anderson

Byte Shop of San Jose
155 Blossom Hill Rd.
San Jose, CA 95123
(408) 226-8383
Larry Grijalva

Byte Shop of Campbell
2626 Union Ave.
San Jose, CA 95124
(408) 377-4685
Ray Lyn

Byte Shop of Sacramento
6041 Greenback Ln.
Citrus Heights, CA 95610
(916) 726-2557
Willard Westphal

COLORADO
Byte Shop/Arapahoe Co.
3464 S. Acoma
Englewood, CO 80110
(303) 761-6232
Bernie Schreibman

Byte Shop of Boulder
2040 30th St.
Boulder, CO 80301
(303) 449-6233
Herman Axelrod

FLORIDA
Byte Shop of Cocoa Bch.
1325 N. Atlantic Ave.
Cocoa Beach, FL 32931
(305) 784-1881
John Dalton

Computer Hut
5905 N.W. 151st St.
Miami Lakes, FL 33014
(305) 821-2667
Donald R. Williams

Byte Shop of Miami
7825 SW 40th St.
Miami, FL 33155
(305) 264-BYTE
Mel Levinson

MicroComputer Systems Inc.
144 So. Dale Mabry Hwy.
Tampa, FL 33609
(813) 879-4301
Forrest K. Hurst

ILLINOIS
The Data Domain
1316 Chicago Ave.
Evanston, IL 60201
(312) 328-6800
Paul Warnke

INDIANA
The Home Computer Store
10447 Chris Drive
Indianapolis, IN 46229
(317) 894-3319
Jim Baughn

The Data Domain
406 S. College Ave.
Bloomington, IN 47401
(812) 334-3607
Boyd R. Borrill

KENTUCKY
The Data Domain
506½ Euclid Ave.
Lexington, KY 40502
(606) 233-3346
Melvyn Cobb

MARYLAND
Computer Workshop
5709 Frederick Ave.
Rockville, MD 20852
(301) 468-0455
John Dorband

MINNESOTA
Byte Shop of Eagan
1434 Yankee Doodle Rd.
Eagan, MN 55121
(612) 452-1841
John Campbell

MISSOURI
Computer Systems Center of St. Louis, Inc.
13461 Olive Blvd.
Chesterfield, MO 63017
(314) 576-5020
H. Joseph Turner, Jr.

Computer Workshop
6903 Blair Road
Kansas City, MO 64152
(816) 741-5055
Terry Smith

NEW HAMPSHIRE
Computer Mart of NH
170 Main St.
Nashua, NH 03060
(603) 883-2386
Ronald J. Cordova

Microcomputers, Inc.
539 Amherst St.
Nashua, NH 03060
(603) 889-1646
Dexter M. Cox

NEW JERSEY
Hoboken Computer Works
No. 20 Hudson Place
Hoboken, NJ 07030
(201) 420-1644

NEW YORK
The Computer Mart of New York
314 Fifth Ave.
New York, NY 10001
(212) 279-1048
Stan Veit

Synchro-Sound Enterprises
193-25 Jamaica Ave.
Hollis, NY 11423
(212) 468-7067
Ray Mustafa

Computer Mart of Long Island
2072 Front St.
East Meadow, NY 11554
(516) 794-0510
Morris Balamut

Byte Shop of Levittown
2721 Hempstead Tpke.
Levittown, NY 11756
(516) 731-8116
John Adams

OKLAHOMA
High Technology
1020 W. Wilshire Blvd.
Oklahoma City, OK 73116
(405) 842-2021
Charles Weddington

OREGON
Byte Shop of Beaverton
3482 SW Cedar Hills Blvd.
Beaverton, OR 97005
(503) 644-2686
Rick Terrell

Byte Shop of Portland
2033 SW 4th St.
Portland, OR 97201
(503) 223-3496
Pat Terrell

PENNSYLVANIA
Byte Shop of Bryn Mawr
1045 W. Lancaster Ave.
Bryn Mawr, PA 19010
(215) 525-7712
Dick Smith

Personal Computer Corp.
Frazer Mall, Rt. 30
Frazer, PA 19355
(215) 647-8460
Walter White

RHODE ISLAND
Computer Power Inc.
M-24 Airport Mall
1800 Post Rd.
Warwick, RI 02886
(401) 738-4477

SOUTH CAROLINA
Byte Shop of Columbia
2018 Green St.
Columbia, SC 29205
(803) 771-7824
Nick Johnson

TENNESSEE
Micro Computer Systems
Route 36, McFee Rd.
Knoxville, TN 37922
(615) 966-9849

UTAH
Byte Shop/Salt Lake City
261 S. State St.
Salt Lake City, UT 84111
Bob Bollinder

VIRGINIA
Shire Enterprises
615 Mitchell St.
Richmond, VA 23222
(804) 321-4560
Sonny Onorati

WISCONSIN
The Milwaukee Computer Store
6916 W. North Ave.
Milwaukee, WI 53213
(414) 259-9140
Denniss Hafemann

CANADA
Trintronics
186 Queen St. West
Toronto, Ontario
Canada, M5V 1Z1
(416) 598-0262
Karen Klein

Byte Shop of Vancouver
2151 Burrard St.
Vancouver 9, B.C.
(604) 736-7221
Marvin McKay

The Pacific Computer Store
4509 - 4511 Rupert St.
Vancouver, B.C. V5R 2J4
(604) 438-DATA
Brian Hyndman

JAPAN
Byte Shop of Tokyo
2-9-9 Sotokanda
Chiyodaku, Tokyo
Kiyotake Ikeda



Ralph was so enthusiastic he hardly even noticed the dog was chewing on one of his program sheets and the kids were making paper airplanes out of others.

Author Ciarcia will not be able to answer any inquiries about the programs described in this article. So don't write!

going to scold first? I just picked up the bottle of Tanqueray from the bar and left the room to go to the shop.

As I descended the stairs into the basement, I listened to the three-ring circus and wondered if children came with volume controls or on/off switches. The sadistic humor ceased when I sat down at my bench and contemplated my next action. I looked at the 8008 system which didn't have BASIC and couldn't play games and shivered a little from the chill of the cold cellar. It took only a few minutes to realize that there was no way I could work downstairs while the gang was upstairs.

I went upstairs and upon entering the den, Ralph said, "I thought you were going to enjoy yourself?"

"I thought so too, Ralph. How are you doing?" I asked trying not to show any frustration.

"I'd say we're almost ready. We entered the player statistics of ten games played last week and the program picked the winner in 40 percent of the cases. I still want to make some other program revisions, but we can do that some other night," he said.

"When we get this program going full blast, we'll have to have better access, though. The program listing the evening's players doesn't come out until one hour before game time, so we'll have to work fast. Do you think we could put it into a motel room near the fronton?" he asked very seriously.

"Not without a great deal of difficulty," I said as calmly as I could, considering the question.

"Well, you know I wouldn't want to put you out. So I guess what we'll have to do is get the player programs of Friday and Saturday evenings, run over here, pump in the data and determine who to bet on. Since the program takes a while to crunch all the numbers and come up with an answer, we'll have to get some other people from the office to help set up a shuttle between here and the fronton to relay the bets and bring back news of any player changes not covered in the initial schedule. It actually won't be too long before we're at that point." Ralph was so enthusiastic he hardly even noticed the dog was chewing on one of his program sheets and the kids were making paper airplanes out of others.

"Wow, did you see that?" I yelled and pointed to the CRT screen.

"What? What happend?" piped Ralph as he squirmed around to check the CRT.

"The CRT momentarily blanked out. I must have a computer problem or the power supply is ready to give out. Hold on, let me

check," I yelled back and dashed out of the room, down the stairs and into the basement. I stood in front of the computer and momentarily watched the twinkling lights. With a grin that only a Cheshire cat could appreciate, I leaned over to the computer and hit the reset button which would cause the monitor upstairs to blank for about a second. I did this twice more for good measure and then pulled the fuse, causing the entire computer to come to a dead stop. I rolled the fuse around in the palm of my hand a few times and gazed at it as I would a gold nugget. When the momentary elation had passed, I pocketed the fuse and went back to the den.

I couldn't have looked more depressed when I arrived upstairs. I just frowned in the direction of Ralph and said in a somber tone, "It died!"

"What do you mean, 'it died?' " was the frantic reply.

"Just that, Ralph. IT DIED! The power supply which I jerry-rigged to add extra memory for you blew up and I don't know what else it took with it. I'm afraid we're out of business for a long time," I added after a deep long audible sigh. "Considering that this is like a child to me, do you think I could be alone, Ralph?" This additional remark was probably a little much, but I was really rolling now.

"Sure. Let me know how things are and whether you need any help. I hate to leave you feeling like this, but it probably is a time you want to be alone." Ralph gathered up his entourage and I waved as they exited, still looking quite somber.

I closed the front door, clapped my hands together and took off for the basement. I had the fuse out of my pocket by the time I reached the fuse box and inserted it immediately. The computer flashed back to life as I placed the blackjack tape I had made into the cassette recorder and hit the play button. After the program was loaded, I took off upstairs with the dog close to my heels. I threw all my computer magazines which I still wanted to read down on the floor around the keyboard and sat down next to it myself, still elated at my triumph. The dog, who had found a comfortable spot in the middle of all the papers, watched as I entered "RUN" on the keyboard.

The program did its thing and displayed a message on the screen. Turning to the dog, I said, "Dog, you show a seven and a jack and the dealer shows a six. Do you want to stand pat or do you want another card?"

"Woof!"■

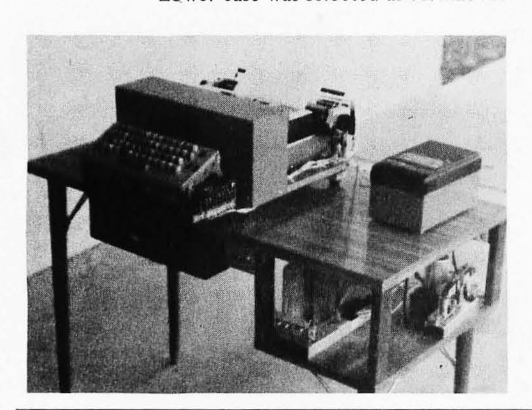
Letters

A SOLENOID TYPEWRITER HACK

In past issues of your excellent magazine I have noted items from your readers expressing interest in converting typewriters for use as printout devices. The accompanying pictures show the results of my efforts along these lines.

The keyboard is a slightly modified Teletype unit *rescued* from a salvage store while the typewriter is a Smith-Corona, Coronet, Super/12 electric portable. The only modification to the typewriter was removal of the carriage-shift-lock-key. The input/output is modified RS232C and operates at about 84 bits per second in ASCII format. The typewriter keyboard actuators are solenoids, also rescued from the salvage store, one per key with logic decoding circuitry and a GI/AY-5-1013 UART.

Lower case was selected as format for letters to obviate a carriage shift for each letter printout. I haven't found the format to be objectionable; the error rate has been acceptable for my hobby use with main maintenance items being ribbon cartridge replacement and contact cleaning on the keyboard. As you can see from the picture, the main effort during construction was function not appearance.



An avid reader of BYTE magazine.

Harold L Baldwin

This avid reader is bashful, but with luck we'll find him sitting down at a typewriter to write an article on his typewriter hack.

PROCESSOR ID NEEDED

I recently bought several PC boards with the intent of removing and reusing the chips. One of these boards is labeled ASSY 591700 Gen Purpose Processor-FBA and it appears to be a complete 16 bit processor. Before dismantling this board, I'd like to find out if it can be used as the basis of a 16 bit mini.

The only clue to the manufacturer seems to be the initials or name LEC and the letters M/L-J etched on the front of the board. I'm not familiar with this company, so any information at all on the board of the manufacturer would be much appreciated.

Larry Ingram
20 Locust St
Cambridge MA 02138

WIDE DISPLAYS IN CHUNKS?

I have just finished reading Dr Robert Suding's piece on television readout devices in the August 76 issue, page 66, and was interested in his discussion of the bandwidth requirements of monitors versus characters per screen.

As part of a project (still in progress) to develop a cheap terminal for use on our campus, I have considered a method of displaying more characters per line that are reasonable on a single monitor by using multiple monitors with a line split between 2 or more monitors. In this scheme, two side by side 6 MHz monitors could be used to present a 64 by 16 format, 3 in a row could display 96 by 16 characters, and a 2 by 2 cluster could go to 64 by 32.

The additional electronics and switching logic needed to support such an increase seems to me almost certain to be far less expensive than going to higher bandwidth monitors whose cost increases very rapidly as large displays are required. (132 by 24 format is not unreasonable to ask for since many printing terminals have such line widths, and once used make a 32 character line extremely restrictive.)

I am not aware of any efforts along these lines; is anyone else?

Robert E Monroe
Associate Director
Computing and Data Processing Center
Wayne State University
Detroit MI 48202

64 by 16 line displays have been demonstrated in one product which has proven itself in the marketplace already, demonstrating at least that resolution capability with a standard television bandwidth. As far as I'm concerned, however, a far better thing to do is to have a multi-window text editor with separate wide displays for different kinds of information in the text file.

For instance, as a documentation machine, consider two side by side displays (64 characters wide or more) with one containing indexing, outline, or

Continued on page 115

And now minis too!
All from a catalog, at discount prices.

Digital's Direct Sales Catalog — the first catalog to offer computer products by mail with off-the-shelf delivery — was such a success, we've come out with an expanded second edition that includes the PDP-8A, the newest member of the world's most popular minicomputer family.

We've also added the LA180 line printer and expanded the sections on microcomputers, logic modules, terminals, cabinets, connectors, accessories, and supplies from the first edition.

Of course, you still get our 5% catalog discount, plus another 4% for cash with your order. You also get a 10-day free trial period, plus our standard 90-warranty. (Since we're selling by mail, you do have to install the equipment yourself.)

If you were among the thousands who ordered our original catalog, you'll get the new one automatically by mail.

If you weren't, why not sign up now?

For your free copy, call 800-225-9480 (Mass. 617-481-7400 ext. 6608). Or write: Components Group, Digital Equipment Corp., MR2-2/M59, One Iron Way, Marlborough, MA 01752. (Catalog sales to U.S. only.)



A Review of Tom Pittman's Tiny BASIC

Richard Rosner
Pocono Rd
Brookfield CT 06804

While at the Personal Computing 76 Trade Show at Atlantic City I bought a copy of Tom Pittman's *Tiny BASIC Interpreter* for my home computer, a 6502 with the MOS Technology TIM program in read only memory. Tom Pittman has several different versions of Tiny BASIC for different systems. I purchased the KIM version even though my computer uses a Teletype and the TIM instead of a keyboard. This allows me to load the interpreter starting at 0200 hexadecimal instead of at 1000 hexadecimal, leaving more memory space for my user programs.

Tiny BASIC is a subset of Dartmouth BASIC. It can handle only 16 bit signed integer numbers between −32768 and +32767 and recognizes the following commands:

CLEAR	REM	GOSUB
INPUT	LET	RETURN
LIST	IF . . . THEN	END
PRINT	GOTO	RUN

The only operations understood by Tiny BASIC are addition, subtraction, multiplication, and division (+, −, *, and /). There are two intrinsic functions. One generates a random number and the other, USR, allows subroutine jumps to machine language programs. More about USR later.

Even with these limitations, Tiny BASIC is very powerful. It is easy to learn and use, and programs can be written very quickly. To use Tiny BASIC you first load the paper tape of the interpreter into the programmable memory. This takes about ten minutes on a Teletype. Then write the addresses of the IO routines into the proper locations and jump to the start of the interpreter. In less than an hour I had it up and running. I wrote and debugged a plot routine one Friday night and made several revisions by the time the weekend was over.

The Tiny BASIC plot program shown in listing 1 accepts the X and Y origin and the X and Y increments as inputs. Then, using a function placed in a subroutine, the program plots a 50 by 50 point graph.

The first input, on line 10, determines the form of the output. A 1 input stops the program after the LIST command at line 20 causes lines 1000 to 1999 to be printed. These lines are the subroutine containing the function to be plotted.

A 2 input causes a request for two more inputs: the X origin and the X increment, T and V in line 200. The printed outputs are the function to be plotted and a table of X and Y values. This table is helpful in choosing the Y origin and Y increment to be used for the graph.

A value of 3 as the first input causes a request for four more inputs: the X origin, the X increment, the Y origin, and the Y increment; T, V, U and W in line 300. The outputs are the function to be plotted and the graph.

Lines 205 through 290 generate and print

```
:LIST

10 INPUTX
20 LIST1000,1999
30 IFX=1GOTO100
40 IFX=2GOTO200
50 IFX=3GOTO300
100 END
200 INPUT T,V
205 X=T
210 PR"X","Y"
215 PR"----","----"
220 GOSUB1000
230 PRX,Y
240 X=X+5*V
250 IFX<T+55*VGOTO220
290 END
300 INPUT T,V,U,W
310 PR
320 PR"  Y =",U,"","",U+25*W,"","",U+50*W
330 PR"","!    .    !    .    !    .    !    .    !"
340 PR"    X=   ---------------------------------------------------"
350 X=T
360 GOSUB1000
365 Y=(Y-U+W/2)/W
370 IFY<0GOTO490
375 IFY=0GOTO500
380 PR"    ";X,"!";
390 IFY<=50GOTO430
400 PR
410 GOTO510
430 Y=Y-1
440 IFY=0GOTO470
450 PR" ";
460 GOTO430
470 PR"+"
480 GOTO510
490 PR"    ";X,"<"
495 GOTO510
500 PR"    ";X,"+"
510 X=X+V
520 IFX<T+51*VGOTO360
590 END
1000 REM THE EQUATION TO BE PLOTTED IS
1010 Y=X*X*X/4-3*X*X+2*X-1
1999 RETURN

:
```

Listing 1: Tiny BASIC listing for a plot routine using a function. The equation to be plotted is in the subroutine located at lines 1000 through 1999. The routine has the ability to print three types of outputs. If a value of 1 is input only the function to be plotted is printed. If a value of 2 is input the program requests the X origin and increment to generate and print a table of X and Y values for the equation on line 1010. The X increment is always 5 times the input increment value. If a value of 3 is input the program requests the X,Y origins and increments. The output consists of the equation to be plotted and the graph.

the table of X and Y values. Starting with the first X, which has the value of T, and for ten other X values spaced at five times the X increment, eleven Y values are calculated by calls to the subroutine at line 1000. In Tiny BASIC, PRINT statements are abbreviated "PR." The PRINT statements at lines 210 and 215 print the heading and the PRINT statement at line 230 prints the X and Y values. The IF statement at line 250 continues the loop until the eleventh X and Y values are calculated and printed.

Lines 310 through 590 generate the graph. The PRINT statements at lines 310 to 340 print the heading. Line 320 prints three values of Y. U is the value of Y at the origin, U + 25 * W is the value of Y at the midpoint of the Y axis, and U + 50 * W is the greatest value of Y plotted. The subroutine call at line 360 brings in a value for Y which is scaled and rounded up at line 365. Lines 370 through 500 print the X value, the X axis, and a "<" or "+" to indicate the value of Y. If the scaled value of Y is less than zero, the IF statement at line 370 causes a jump to the PRINT statement at line 490 where a caret is printed on the X axis to indicate that the value of Y is below the lower limit of the Y axis. If the scaled value of Y equals zero, the IF statement at line 375 causes a jump to line 500 where a plus sign is printed on the X axis. If the scaled value of Y is greater than zero, the PRINT at line 380 prints the value of X and an exclamation mark for the X axis. The semicolon at this PRINT statement causes the next PRINT statement to write on the same line without any spacing. If the scaled value of Y is greater than 50 the PRINT at line 400 closes this line and causes the next PRINT statement to start printing at the beginning of the next line. If the scaled value of Y is between 1 and 50 the statements on lines 430 through 470 cause a plus sign to be printed Y spaces above the X axis. The statements at lines 510 and 520 increment X and continue the loop until the last (51st) values of X and Y are computed and plotted.

The example of listings 2 and 3 are for the cubic equation:

$$Y = \frac{X^3}{4} - 3X^2 + 2X - 1$$

Listing 2a is the table of values for X and Y starting at X equal to −50 with an X increment of 2. The jump to negative values of Y (at X equal to 40) and the exceptional values of Y at X equals −50 and −40 are due to computations exceeding the range of Tiny BASIC. Listing 2b is the same table with the first term rearranged. This keeps the value of Y within the range of precision for variables for a larger range of X. Listing 3 is the plot of the equation with the starting value of X equal to −25, the X increment equal to 1, and the starting value of Y and the Y increment equal to −100 and 4, respectively.

```
:RUN 2, -50, 2

1000 REM THE EQUATION TO BE PLOTTED IS
1010 Y=X*X*X/4-3*X*X+2*X-1
1999 RETURN

X         Y
----      ----
-50       -6083
-40       -4497
-30       -9511
-20       -3241
-10       -571
0         -1
10        -31
20        839
30        4109
40        -5105
50        -8919

:
```

*Listing 2a: The 2 command has been read telling the program to request a value for the X origin and increment. A table of X and Y values for the function of line 1010 is also output. In this example the X origin is —50 and the X increment is 5*2. For the X values of —50, —40, 40 and 50 the incorrect Y value is given. This is because the value that is computed is outside the range of the Tiny BASIC's capabilities. After looking at this table the origins of the program can be changed so that values capable of being used are generated. Another use of the table is to determine the values of the Y axis.*

A small program change in lines 1000 to 1999 allows discontinuous functions to be generated and plotted as in listing 4. These lines are the ones that contain the function to be plotted. Instead of using a simple expression, conditions are set up to determine the starting Y value of the graph on line 1060, the upper and lower limits between which the graph will oscillate on lines 1020 and 1000, and the risetime and falltime of the graph on lines 1040 and 1051. By examining lines 1000 to 1999 in listing 4 it can be seen that the graph is independent of X but is a function of its own previous value.

If you want to use the entire graphing routine as a subroutine you may want to renumber each line. Be sure that there are no statement number conflicts with your routines and that all GOTO, GOSUB, and LIST statements are properly changed. As an alternative you can use statement numbers from 2000 through 32767 and put a "GOTO (first line number of your routine)" at statement number 1.

You will notice that no unnecessary blanks are included within each statement. This makes it slightly more difficult to read the program but does conserve space. Each blank requires a byte of storage. At an average of 2 blanks per line, this saves 90 bytes. As shown in listing 1, the program requires about 650 bytes of programmable memory storage.

The USR instruction allows a Tiny BASIC program access to machine language subroutines and to the rest of the memory space including IO ports. The format for using USR is:

D = USR (P, L, A)

where P is the address of the machine language subroutine, L is a variable passed through the X and Y processor registers, and A is a variable passed through the processor's accumulator. Upon exiting from such a user subroutine and returning to Tiny BASIC, the high order eight bits of D are taken from the Y register and the low order eight bits of D are taken from the accumulator. The D, P, L and A values are in decimal.

Two subroutines addressable by the USR function are included in Tiny BASIC. These are the PEEK and POKE routines. The PEEK routine reads the byte at location L, converts this byte to its decimal equivalent and sets D to this value. The POKE routine loads the low order eight bits of A into the memory location specified by L. The PEEK routine is located at P = 532 for the KIM version and the POKE routine is located at decimal address P = 536. The addresses for the TIM version are decimal addresses P = 4116 for PEEK and P = 4120 for POKE. For the 6800 these addresses are P = 276 for PEEK and P = 280 for POKE.

With these routines you can read and write into your computer's IO ports. Set L to the port address, decimal, P to the PEEK address and execute

D = USR (P, L).

```
:RUN 2, -50, 2

1000 REM THE EQUATION TO BE PLOTTED IS
1010 Y=X*X/4*X-3*X*X+2*X-1
1999 RETURN

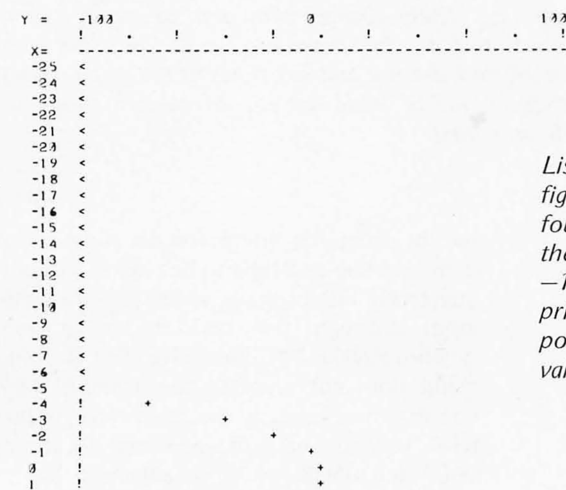
X         Y
----      ----
-50       26685
-40       -20881
-30       -9511
-20       -3241
-10       -571
0         -1
10        -31
20        839
30        4109
40        11279
50        23849

:
```

Listing 2b: This is a run of the program under the 2 command using the same origin and increment values as in listing 2a. The function to be graphed, however, is still a cubic. Rearranging the order of operations helps avoid overflow errors. With this equation the only value that falls outside the ability of Tiny BASIC appears when X = —50.

```
:RUN 3, -25, 1, -100, 4

1000 REM THE EQUATION TO BE PLOTTED IS
1010 Y=X*X*X/4-3*X*X+2*X-1
1100 RETURN
```



Listing 3: The plot of the cubic equation whose table of values is shown in figure 2a. The graduations on the Y axis are from −100 to +100. A caret is found on the X axis from the values of −25 to −5. This mark indicates that the point on the graph is less than the Y origin, in other words is less than −100 for this example. If the value of Y is equal to 0, a plus sign, +, will be printed on the X axis. If the value of Y is greater than 0, an exclamation point, !, will be printed on the X axis, and a plus sign will be printed at the value.

The port data is loaded into D. Set L to the decimal port address, P to the POKE address, A to the value to be transferred and execute

D = USR (P, L, A).

This causes the value of A to be written into the port.

By using the PB7 pin on a 6530, very accurate pulses from two microseconds to about a quarter of a second in width can be generated directly from Tiny BASIC. If L is set to the base address of the 6530 timer, then the addresses L+12, L+13, L+14 and L+15 determine a factor by which A is multiplied to get the pulse width. These factors are:

Address	Pulse Width
L + 12	1 * A
L + 13	8 * A
L + 14	64 * A
L + 15	1024 * A

If the 6502 is using a one megahertz clock, executing the line

D = USR (536, L + 13,216)

causes a pulse of width 1728 µs (8 * 216) to be generated at PB7. PB7 must be set up as an input (not output) and the 6530 timer must be dedicated to this function.

To return to the TIM monitor the Tiny BASIC program should execute this sequence of statements:

D = USR (536, 0, 0)
D = USR (536, 1, 96)
D = USR (0)

The first instruction loads 00, the 6502 BRK software interrupt instruction, into location 0000. The second instruction in the above sequence loads 96, 60 hexadecimal, the 6502 RTS return from subroutine instruction, into location 0001. The last instruction causes Tiny BASIC to jump to the subroutine at 0000 where the instruction executed by the 6502 causes the TIM monitor to take over. To return to Tiny BASIC, just have the TIM monitor execute the instruction at 0001 by typing "G" at the console. If the sequence of instructions above was executed from a Tiny BASIC program by preceding each instruction with a line number and commanding RUN, the next instruction in the user's Tiny BASIC program will be executed when returning from the TIM monitor.

Saving and reloading programs from paper tape or magnetic tape is easy with Tiny BASIC if you have a debugged version of the interpreter. The procedure is to save the program by typing LIST, turning on the paper tape punch, and pressing the RETURN key on the Teletype. To reload the program press LINE FEED, load the paper tape into the tape reader and turn the reader on. The LINE FEED command suppresses

```
:RUN 3, 0, 1, 0, 1

1000 L=25
1010 REM L IS THE LOWER LIMIT
1020 M=45
1030 REM M IS THE UPPER LIMIT
1040 S=4
1050 REM S IS THE RISETIME
1351 REM  THE FALLTIME IS  1/(M-L)
1060 K=12
1070 REM K IS THE STARTING Y VALUE
1080 IFX=T Z=(K-L)/S
1090 Y=S*Z+L
1100 Z=Z+1
1110 IFY>=M Z=0
1999 RETURN
```

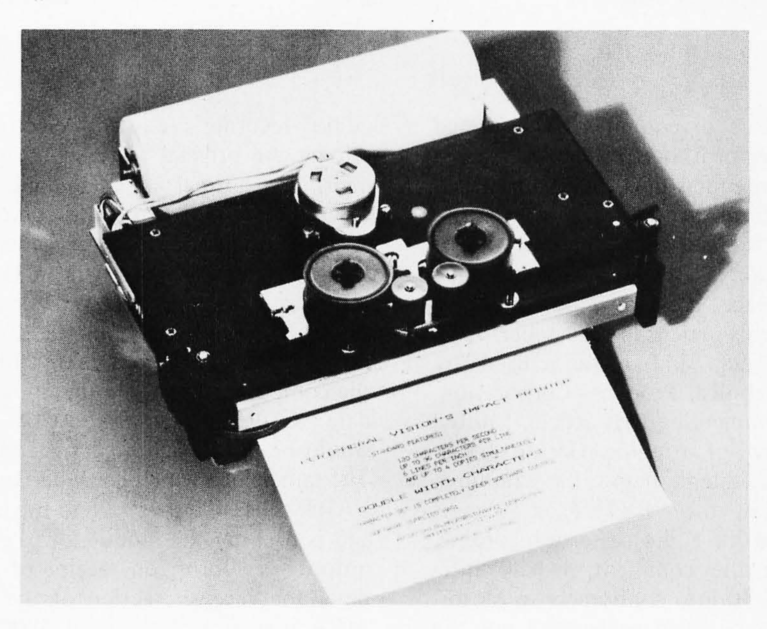
Listing 4: Plot of a sawtooth function. The function is independent of X but is dependent on the previous value of Y. This type of graph may be plotted by changing lines 1000 through 1999 from those shown in listing 1 to those shown at the top of this listing. These changes allow you to specify the Y value that the graph should start at, the upper and lower limits between which the graph will oscillate, and the rise and fall times of the graph. Using this type of evaluation technique a wide variety of discontinuous or oscillating functions can be generated.

```
Y =   0                       25                       50
      !    .    !    .    !    .    !    .    !    .    !
X=    ---------------------------------------------------
0     !            +
1     !                +
2     !                    +
3     !                        +
4     !                            +
5     !                                +
6     !                                    +
7     !                                        +
8     !                                            +
9     !                        +
10    !                            +
11    !                                +
12    !                                    +
13    !                                        +
14    !                                            +
15    !                        +
16    !                            +
17    !                                +
18    !                                    +
19    !                                        +
20    !                                            +
21    !                        +
22    !                            +
23    !                                +
24    !                                    +
25    !                                        +
26    !                                            +
27    !                        +
28    !                            +
29    !                                +
30    !                                    +
31    !                                        +
32    !                                            +
33    !                        +
34    !                            +
35    !                                +
36    !                                    +
37    !                                        +
38    !                                            +
39    !                        +
40    !                            +
41    !                                +
42    !                                    +
43    !                                        +
44    !                                            +
45    !                        +
46    !                            +
47    !                                +
48    !                                    +
49    !                                        +
50    !                                            +
```

output from the interpreter to allow more than one line at a time to be read in without interfering line prompts which can cause the input through the tape reader to lose synchronization. If the LINE FEED command does not suppress line prompts then you may have a bug in the interpreter. In the KIM version check hexadecimal locations 06C7 and 06CB for 13 hexadecimal, X off, and 0A hexadecimal, LINE FEED, respectively. These bytes may be reversed. If they are, switch them and try again. In the TIM version check locations 14C7 and 14CB. In both cases the lower location should contain 13 hexadecimal and the higher location should contain 0A hexadecimal. Only tapes shipped before September 1976 should have this problem.

To obtain a copy of the Tiny BASIC interpreter for your machine, send a description of your system along with a check for $5 to Tom Pittman, itty bitty computers, POB 23189, San Jose CA 95153. In return you will receive the interpreter on a paper tape and a user's manual. The 6502 microprocessor, TIM, KIM, and 6530 are available from MOS Technology Inc, Morristown PA, as is a line of KIM-n products which can be assembled into a complete computer.■

Peripheral Vision impacts your computer.



WITH A FULL-SIZE, LOW-COST IMPACT PRINTER.

Until now, the hobbyist and small businessman have had one major problem in assembling a reasonably price microprocessor system with the capabilities found in the more costly computers. It was impossible to find a high-quality, high-output printer for hard copy needs at an affordable price.

Peripheral Vision has come up with a solution.

We are offering a full-size *impact* printer designed for microprocessors—and it comes with a mini price. Kit prices start as low as $495 for the printer and interface card. And that won't impact your pocketbook.

Peripheral Vision's printer is loaded with capabilities. Take a look:

- It's fast—120 characters per second
- 96 characters per line, 12 characters per inch horizontal, 6 lines per inch
- Makes up to 4 copies simultaneously
- Character set and pitch variable under software control
- 5 x 7 character matrix
- Ribbon has built-in re-inkers for a life of 10,000,000 characters
- Paper can be either a standard 8½-inch roll, fanfold or cut page
- Interfaces to 8-bit parallel ports

Just remember, Peripheral Vision is committed to helping you get along with your computer. The new printer we are offering is another example. It is high quality, low in cost and will definitely impact your system.

Write or call now to find out how to impact your computer.



P.O. Box 6267 Denver, Colorado 80206 (303) 777-4292

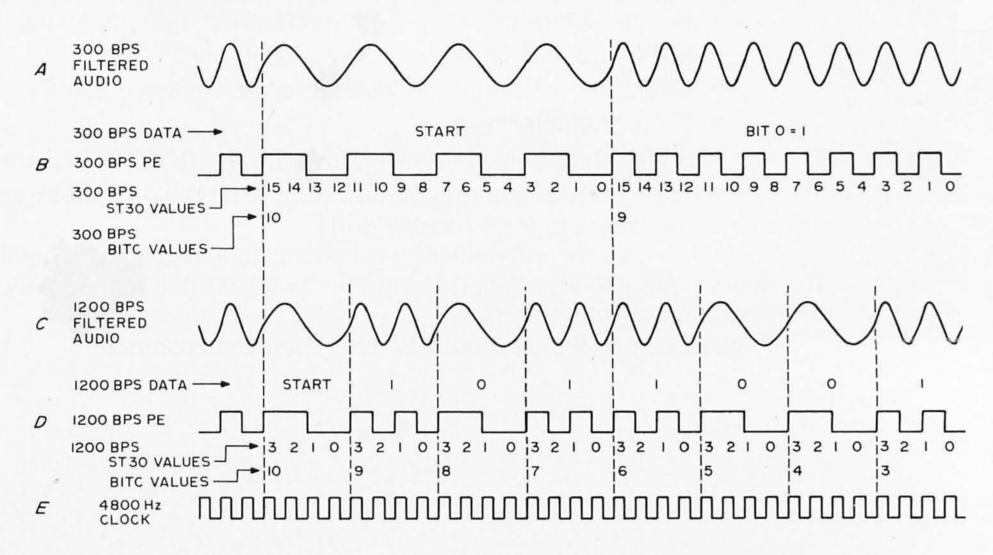
A Software Controlled 1200 bps Audio Tape Interface

Carl Helmers
Editor

Often a piece of equipment is designed which will perform perfectly well in the mode of its intended operation, but which can also perform in ways not necessarily intended by the designer. An example of such a situation is the AC-30 audio tape cassette interface subsystem described by Gary Kay in his article in December 1976 BYTE, page 98, and manufactured by Southwest Technical Products Corporation. In a box accompanying that article, I noted the possibility of running Gary's design considerably faster than the nominal 300 bps rate of the "BYTE" or "Kansas City" standard for audio cassette recording. Since making the comment, I have integrated the AC-30 into my homebrew Motorola 6800 system and have worked out the details of this high speed use of the AC-30.

The Kansas City standard uses an AC coupled redundant phase encoded method of recording which takes advantage of the audio recorder's existing electronics. Retaining the original audio electronics allows continued normal use of the audio medium when you're not using the computer. This type of modulation is often generated from a 4800 Hz clock source and a UART (universal asynchronous receiver-transmitter) or ACIA (asynchronous communications interface adapter) integrated circuit interfaced to the computer's bus to do the shifting out of data. Gary assumed an ACIA or UART would be used when he designed the AC-30. The same clock which is divided by 16 in the UART shifts the data out to the modulator, and is used by the modulator to generate an output waveform; on receipt of data from tape, the receiver section of the UART or ACIA uses a clock at about 4800 Hz which is derived from the tape data. The logic in the AC-30 which is used to generate and decode the waveform is only dependent upon the 1200 Hz and 2400 Hz frequencies

Figure 1: Timing of various interface signals. This diagram shows audio and TTL logic level signals for 300 bps and 1200 bps data referenced to the same 4800 Hz clock shown at E. On output, the timing of the clock governs generation of the waveforms, and on input the waveforms of A or C are used to regenerate clock data which averages 4800 Hz but with some jitter. The waveforms of A and B, respectively, show filtered audio and equivalent redundant Manchester I encoded digital data for a



which are derived from the 4800 Hz output clock by frequency division in toggle flip flops.

The fact that it is only the 1200 and 2400 Hz frequencies on the tape which count when modulating or demodulating the data is what makes this trick of 1200 bps recording with the AC-30 possible. Figure 1 shows the typical waveforms of the redundant phase encoded format at the beginning of a 300 bps character transmission from the UART or ACIA, and for an entire character transmitted by a suitable algorithm in the asynchronous data format at 1200 bps. For 300 bps transmissions, a logical 0 bit in the tape record consists of four complete cycles of 1200 Hz and a logical 1 bit consists of eight complete cycles of 2400 Hz. The switching between the two frequencies is synchronous with the zero crossings so that when the digital square wave signals generated by the logic are filtered in the AC-30, and by the act of recording on magnetic tape, the results coming back into the demodulator are smooth sine wave approximations (see figure 1A for the 300 bps version). In the 1200 bps version, we eliminate most of the redundancy from the redundant Manchester code by using only one complete cycle of 1200 Hz for a logical 0 bit, and two complete cycles of 2400 Hz for a logical 1 bit. If we were to use a true Manchester I phase encoding method (see Ira Rampil and Jack Bremier's article in March 1977 BYTE, page 38), a 2400 bps rate would be possible by eliminating one more level of redundancy. But the design of the AC-30 prevents generation of the true phase encoded signal and the goal was to use the AC-30 without any hardware modifications in its circuit.

With this analysis of the situation, and a mental picture of what is seen in figure 1, I set out to make an interface which would drive the AC-30 at 1200 bps. Since I am a software person at heart, and since I wanted to allow the possibility of eventually dispensing with the asynchronous communications format by means of a new program design, I decided to leave out the UART or ACIA chips and make a simple hardware interface for a programmed data format driven by interrupts.

The Hardware: A Dedicated Port and Real Time Clock

In order to do a software simulation of a data transmission format, some form of timing must be generated, both as the 4800 Hz clock source for output through the AC-30 and to synchronize data with that clock or the tape derived clock on input. Within the context of my own system, and bearing in mind more general uses of the technique as well, I chose to use a separate timing source driving the processor's nonmaskable interrupt (NMI) line as the means of synchronizing the software to the time base. On output, the NMI line would have to be driven by a 4800 Hz clock source; on input the NMI line would have to be driven by the derived clock provided by the AC-30.

The output clock source is a simple 555 oscillator, IC9 in figure 2. This was set up with a variable resistor and a timing capacitor such that 4800 Hz was well within the

300 bps signal. The time ordered sequence of data starts with a "start" bit followed by several data bits. The waveforms of C and D show a complete character transmission at 1200 bps both as filtered audio (C) and redundant Manchester I encoded digital data (D). Above B and D are shown verbal descriptions of the data being transferred. Noted below signals B and D are the values of the variables ST30 and BITC during output operations (see listing 1 and procedure 6).

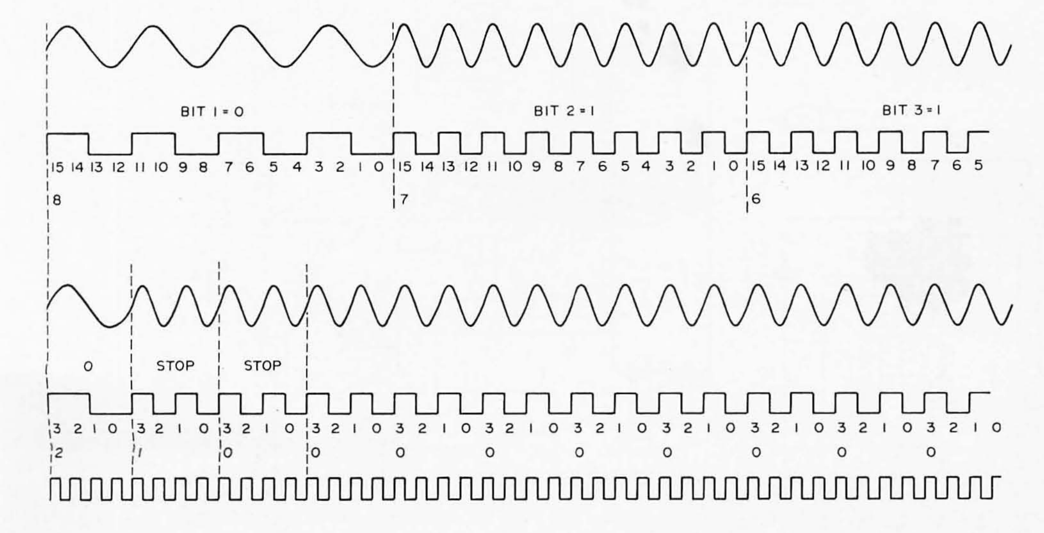
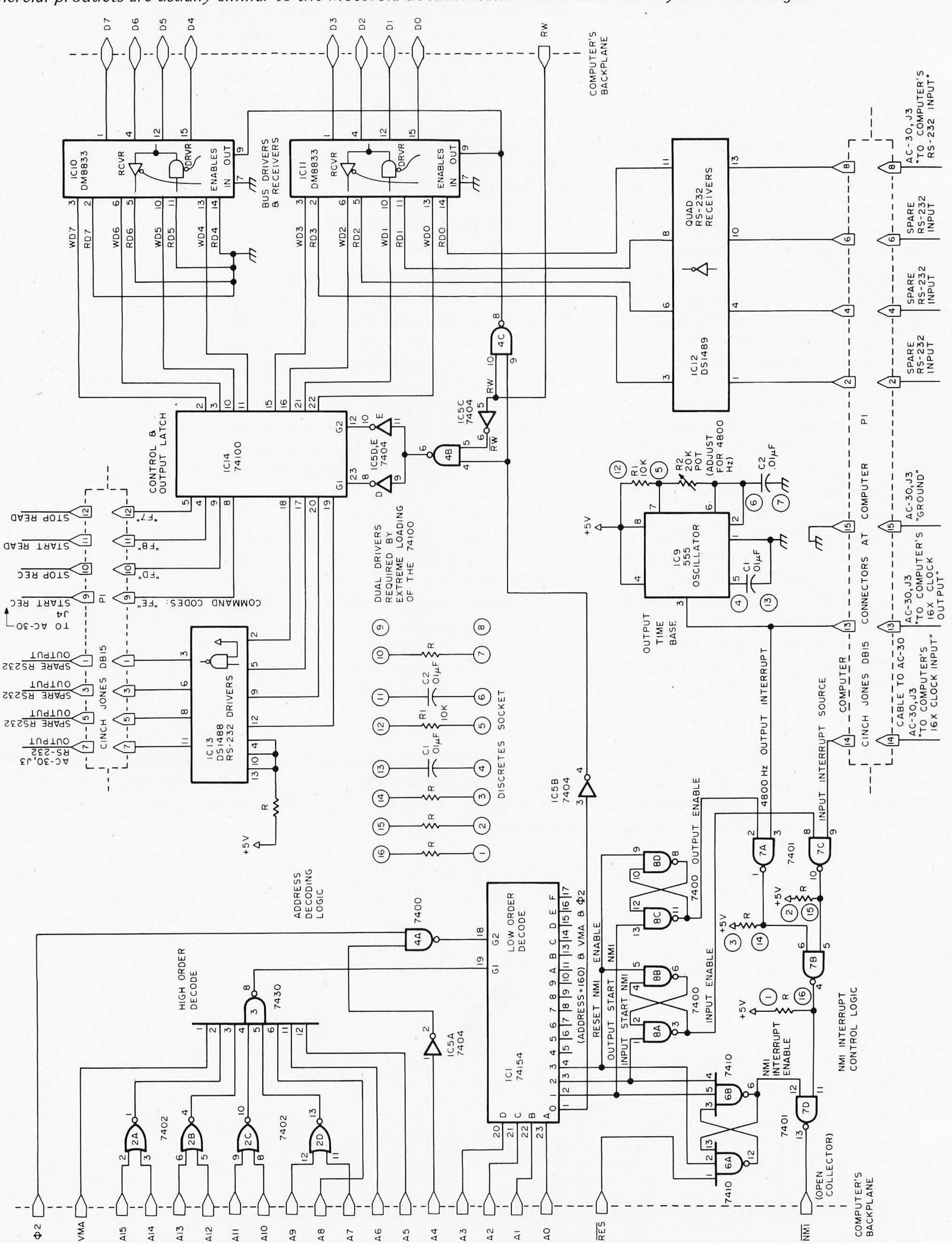


Figure 2: The hardware needed to add this general purpose high speed AC-30 interface. The logic includes address decoding for the addresses in listing 1 and table 1, latched output from location 160 for control and data transmission purposes, drivers and receivers for RS-232 lines, an oscillator time base for output interrupts, as well as interrupt control logic and an open collector driver for the nonmaskable interrupt line. For those who use a commercial 6800 system, the bus pins should be allocated to the equivalent backplane pins of your system, and address decoding should be altered to fit your system. The names used in commercial products are usually similar to the Motorola documentation from which this system was designed.



tuning range. The .01 μF disk ceramic capacitor C2 which I used as part of the frequency determination network for the oscillator may not have been the best choice for insensitivity to temperature and long term frequency stability, but previous experience with 555 oscillators of this type proved stable over the course of nearly a year of use in a slower tape interface which had been my primary mass storage method until now.

After the oscillator is wired, it can be tuned to 4800 Hz by several methods. One method is to borrow a friend's frequency meter, or use an oscilloscope with presumably accurate calibration of its time base. I used a simple interrupt driven program in the processor to count the number of processor cycles between clock interrupts since I don't trust my oscilloscope's calibration and don't have a frequency counter, but do have a crystal controlled processor clock.

Listing 1: An assembly of the AC-30 interface routines. The origin of this assembly was chosen arbitrarily as hexadecimal 2000 to simplify relocation. In actual use, the routines were assembled to end up at the top of the previous code limits for the author's IMP 6800 monitor and text editor programs. (IMP, which resides in write protected memory at locations 1000 to 1FFF, stands for "Interactive Manipulator Program.") The addresses of FLAG, MEM and ERRS used here are set by the structure of IMP — as arguments in different versions of this set of routines, any old address would do. The IO address locations reflect the hardware of figure 1, and MILI is a one millisecond delay loop program resident in a random piece of ROM which fills up part of the bootstrap ROM of the system. In the system in which this algorithm was used, the NMI (and other interrupts) vector to jump instructions in programmable memory in the first 12 bytes of address space.

Number	Type	+5 V	GND	−12 V	+12 V
IC1	74154	24	12		
IC2	7402	14	7		
IC3	7430	14	7		
IC4	7400	14	7		
IC5	7404	14	7		
IC6	7410	14	7		
IC7	7401	14	7		
IC8	7400	14	7		
IC9	555	8,4	1		
IC10	DM8833	16	8		
IC11	DM8833	16	8		
IC12	DS1489	14	7		
IC13	DS1488	—	7	1	14
IC14	74100	24	7		

```
LOCN B1 B2 B2
2000           > ORG $2000
2000           >*
2000           >*       AC-30 INTERFACE ROUTINES
2000           >*
2000           >* "BYTE STANDARD" INTERFACE SOFTWARE
2000           >*      ALSO CALLED "KANSAS CITY", "CUTS" FORMAT
2000           >*      BUT A ROSE BY ANY OTHER NAME STILL SMELLS
2000           >*
2000           >* DESIGN BY CARL HELMERS, EDITOR, BYTE
2000           >*
2000           >*     GENERAL AC-30 IO DESIGNED DECEMBER 26 1976
2000           >*     TAPE 200A LOCATION 020 FILE="AC-30 IO"
2000           >*
2000           >*
2000           >* THIS SET OF ROUTINES RUNS AT 300 OR 1200 BPS
2000           >*      AT 300 BPS, IS "BYTE","KANSAS CITY" OR "CUTS" FORMAT
2000           >*      AT 1200 BPS, IS 1200 BPS REDUNDANT MANCHESTER I
2000           >*           PHASE ENCODING
2000           >* THE VARIABLE "RATE" DETERMINES SPEED OF DATA
2000           >*
2000           >* EQUATES
0050           >ST30 EQU $50 OUTPUT BIT STATE
0051           >BITV EQU $51 OUTPUT BIT VALUE (0 OR 1)
0052           >BITC EQU $52 OUTPUT BIT COUNT
0053           >DATL EQU $53 LEFT DATA & TRAILER
0054           >DATR EQU $54 RIGHT DATA & START BIT
2000           >*    IN ARTICLE, "DAT" REFERS TO BOTH OF ABOVE
0055           >NXDA EQU $55 NEXT DATA (2 BYTES)
0057           >DELA EQU $57 DELAY COUNT (2 BYTES)
0059           >RATE EQU $59 DATA RATE
2000           >*      300 BPS VALUE=16     1200 BPS VALUE=4
005A           >HALF EQU $5A HALF RATE
0160           >AC30 EQU $160 AC30 DATA OUTPUT AND INPUT ADDRESS
0161           >ACGO EQU $161 OUTPUT NMI INTERRUPT ENABLE ADDRESS
0163           >ACOF EQU $163 NMI INTERRUPT DISABLE ADDRESS
0138           >MILI EQU $138 MILI(B) - B MILLISECOND DELAY
2000           >*           MILI IS A TIMING LOOP IN MONITOR
0004           >NMIV EQU $4 NMI ADDRESS VECTOR
2000           >*  JMP OP CODE AT LOCATION $3 ALLOWS WRITABLE VECTOR
002E           >FLAG EQU $2E MONITOR'S INPUT OR OUTPUT TRANSFER FLAG
0015           >MEM  EQU $15 MONITOR'S EVOLVING DATA POINTER
0033           >ERRS EQU $33 MONITOR'S ERROR COUNT FIELD IN VERIFY
2000           >*
2000           >* THE FOLLOWING JUMPS SIMPLIFY PATCHING INTO MONITOR
2000           >* DURING INITIAL TESTING BY PROVIDING FIXED ENTRIES
2000 7E 00 00 >OPE1  JMP  U1OP
2003 7E 00 00 >READ  JMP  U1RD
2006 7E 00 00 >WRIT  JMP  U1WR
2009 7E 00 00 >VRFY  JMP  U1VY
200C 7E 00 00 >CLOS  JMP  U1CL
200F 7E 00 00 >OPE2  JMP  U2OP
2012           >*
2012           >*  AC-30  NMI  INTERRUPT HANDLER
2012           >CY30  EQU   *
2012 96 51    >       LDAA  BITV
2014 8A FE    >       ORAA  #$FE
2016 B7 01 60 >       STAA  AC30
2019 7A 00 50 >       DEC   ST30
201C 27 00    >       BEQ   NEXB
201E 3B       >       RTI
201F 4F       >NEXB  CLRA
201D 01
2020 77 00 53 >       ASR   DATL
2023 76 00 54 >       ROR   DATR
2026 24 00    >       BCC   CYZN
2028 4C       >       INCA
2029 97 51    >CYZN  STAA  BITV
2027 01
202B 96 52    >       LDAA  BITC
202D 27 00    >       BEQ   CYZR
202F 4A       >       DECA
2030 97 52    >CYZR  STAA  BITC
202E 01
2032 96 59    >       LDAA  RATE
2034 97 50    >       STAA  ST30
2036 3B       >       RTI
2037           >*
2037           >*
2037           >*  AC-30 MONITOR ROUTINES
2037           >*
2037           >*
2037           >* WRITE AC-30 300 BPS
2037           >U1WR  EQU   *
2007 20 37
2037 96 52    >       LDAA  BITC IF BIT COUNT > 0 THEN WAIT
2039 26 FC    >       BNE   U1WR
203B DE 55    >       LDX   NXDA
203D DF 53    >       STX   DATL := NXDA
203F 86 0B    >       LDAA  #11
2041 97 52    >       STAA  BITC := 11 [SET BIT COUNT];
2043 86 FF    >       LDAA  #$FF
2045 97 55    >       STAA  NXDA [SET TRAILING BIT PATTERN]
2047 DE 15    >       LDX   MEM
2049 A6 00    >       LDAA  0,X
204B 97 56    >       STAA  NXDA+1 := @MEM [DEFINE NEXT OUTPUT]
204D 78 00 56 >       ASL   NXDA+1
2050 79 00 55 >       ROL   NXDA := SHL(NXDA) [INSERT START BIT]
2053 39       >       RTS
2054           >*
2054           >* 1200 BPS OPEN
2054 86 04    >U2OP  LDAA  #4
2010 20 54
2056 97 59    >       STAA  RATE
2058 20 00    >       BRA   UUOP
205A           >*
205A           >* 300 BPS OPEN
205A 86 10    >U1OP  LDAA  #16
2001 20 5A
205C 97 59    >       STAA  RATE
205E 96 59    >UUOP  LDAA  RATE
2059 04
2060 47       >       ASRA
2061 97 5A    >       STAA  HALF
```

Listing 1, continued:

```
2063 7D 00 2E >       TST   FLAG IS IT OUTPUT?
2066 26 00    >       BNE   U100
2068 7E 00 00 >       JMP   U1I0 IF NOT THEN INPUT OPEN
206B          >*
206B          >* OUTPUT OPEN
206B 86 EF    >U100   LDAA  #$EF TURN
2067 03
206D B7 01 60 >       STAA  AC30 ON
2070 86 FF    >       LDAA  #$FF THE
2072 B7 01 60 >       STAA  AC30 MOTOR;
2075 CE C3 50 >       LDX   #9999+9999+9999+9999+9999+5  (IE: 50,000)
2078 DF 57    >       STX   DELA
207A CE FF FF >       LDX   #$FFFF
207D DF 53    >       STX   DATL
207F DF 55    >       STX   NXDA
2081 96 59    >       LDAA  RATE
2083 97 50    >       STAA  ST30
2085 97 52    >       STAA  BITC
2087 CE 00 00 >       LDX   #U101 SET UP INTERRUPTS
208A DF 04    >       STX   NMIV
208C 7F 00 51 >       CLR   BITV
208F 4F       >       CLRA
2090 7F 01 61 >       CLR   ACGO
2093 4D       >U104   TSTA
2094 27 FD    >       BEQ   U104
2096 39       >       RTS
2097          >*
2097          >* OUTPUT OPEN INTERRUPT
2097 DE 57    >U101   LDX   DELA
2088 20 97
2099 09       >       DEX
209A 27 00    >       BEQ   GONE
209C DF 57    >       STX   DELA
209E 3B       >       RTI
209F          >GONE   EQU   *
209B 03
209F CE 20 12 >       LDX   #CY30
20A2 DF 04    >       STX   NMIV
20A4 86 01    >       LDAA  #1
20A6 97 51    >       STAA  BITV
20A8 30       >       TSX
20A9 63 02    >       COM   2,X  LEAVE OPEN HANGUP
20AB 3B       >       RTI
20AC          >*
20AC          >* 300 BPS CLOSE
20AC 7D 00 2E >U1CL   TST   FLAG
200D 20 AC
20AF 26 00    >       BNE   U10C
20B1 7E 00 00 >       JMP   U1IC
20B4          >*
20B4          >* OUTPUT CLOSE
20B4 96 52    >U10C   LDAA  BITC
20B0 03
20B6 26 FC    >       BNE   U10C WAIT FOR END LAST CHR
20B8 DE 55    >       LDX   NXDA
20BA DF 53    >       STX   DATL
20BC DF 55    >       STX   NXDA
20BE CE 00 6E >       LDX   #110
20C1 DF 55    >       STX   NXDA
20C3 86 0B    >U102   LDAA  #11
20C5 97 52    >       STAA  BITC
20C7 96 52    >U103   LDAA  BITC
20C9 26 FC    >       BNE   U103
20CB DE 55    >       LDX   NXDA
20CD 09       >       DEX
20CE DF 55    >       STX   NXDA
20D0 26 F1    >       BNE   U102
20D2 7F 01 63 >       CLR   ACOF TURN OFF NMI
20D5 86 DF    >       LDAA  #$DF
20D7 B7 01 60 >       STAA  AC30  TURN OFF
20DA 86 FF    >       LDAA  #$FF  THE
20DC B7 01 60 >       STAA  AC30  MOTOR
20DF C6 FA    >       LDAB  #250
20E1 37       >       PSHB
20E2 BD 01 38 >       JSR   MILI
20E5 33       >       PULB
20E6 BD 01 38 >       JSR   MILI
20E9 39       >       RTS
20EA          >*
20EA          >* INPUT OPEN
20EA 86 BF    >U1I0   LDAA  #$BF
2069 20 EA
20EC B7 01 60 >       STAA  AC30
20EF 86 FF    >       LDAA  #$FF
20F1 B7 01 60 >       STAA  AC30
20F4 C6 FA    >       LDAB  #250
20F6 7F 00 51 >       CLR   BITV
20F9 37       >       PSHB
20FA BD 01 38 >       JSR   MILI
20FD 33       >       PULB
20FE BD 01 38 >       JSR   MILI
2101 CE 00 00 >       LDX   #U1I1
2104 DF 04    >       STX   NMIV
2106 7F 01 62 >       CLR   ACGO+1
2109 39       >       RTS
210A          >* AWAIT START
210A B6 01 60 >U1I1   LDAA  AC30
2102 21 0A
210D 84 01    >       ANDA  #1
210F 26 00    >       BNE   U1IA
2111 96 5A    >       LDAA  HALF BIT TIME
2113 97 50    >       STAA  ST30
2115 CE 00 00 >       LDX   #U1I2
2118 DF 04    >       STX   NMIV
211A 3B       >U1IA   RTI
2110 09
211B          >*
211B          >* VERIFY START
211B B6 01 60 >U1I2   LDAA  AC30
2116 21 1B
```

Masking the Unmaskable

Driving the NMI line for interrupts has one potential problem with a 6800 processor: The hardware of the microprocessor itself has no way to turn off the source of the interrupts. When I am not using the tape interface, I don't want to have any interrupts. When I am listening to input data, I don't want interrupts from the output clock. When I am recording the data, I don't want any spurious interrupts from the input clock logic of the AC-30. Thus some switching and masking logic is in order.

In figure 2, IC1, IC2 and IC3 are used together to decode several addresses by combining selection of the address with VMA (valid memory address) and the processor's Φ_2 signal. Three of these addresses, hexadecimal 0161, 0162 and 1063, are not

```
***************************************************************
      NOTES ON USE OF THE AC-30 WITH SUPERSCOPE MODEL C-104
***************************************************************

* TEMPERATURE SENSITIVITY

   WHEN USED IN A COLD BASEMENT (ESTIMATED TEMPERATURE
OF 55-60 DEGREES FAHRENHEIT) THE RECORDS ON TAPE WERE
OFTEN UNREADABLE.  THE SUSPECTED CAUSE OF THIS
PHENOMENON IS CHANGES IN LINEAR DIMENSIONS OF TAPE, SINCE
ADJUSTING THE TAPE SPEED TO NON STANDARD VALUES CURED
THE PROBLEM.  THE ADJUSTMENT WHICH WORKED WAS ABOUT 1-2%
SLOWER THAN NORMAL BY THE C-104'S DIAL, WHICH IS CONSISTENT
WITH A CONTRACTION OF THE LINEAR DIMENSIONS OF THE TAPE
DUE TO THE LOWER SYSTEM TEMPERATURE.


* SENSITIVITY TO "TONE" AND "VOLUME" CONTROLS

   THE PLAYBACK RELIABILITY IS EFFECTIVELY INDEPENDENT
OF THE VOLUME OF PLAYBACK OVER A RANGE COVERING THE
CENTER OF THE VOLUME SCALE OF THE RECORDER (3 TO 8 ON
ITS ARBITRARY CALIBRATION SCALE).  HOWEVER THERE WAS
CONSIDERABLE SENSITIVITY TO THE TONE CONTROL SETTING,
WITH THE OPTIMAL CONTROL BEING TOWARD THE "-" SIDE OF
0 IN THE C-104'S RANGE OF  -5 TO +5.


GENERAL SUMMARY OF RESULTS AS OF JANUARY 27 1977...

   AFTER ONE MONTH OF USE, THE INTERFACE HAS WORKED
RELIABLY AS THE PRINCIPAL MASS STORAGE DEVICE IN MY SYSTEM
AFTER IT WAS ASSEMBLED AT A LOCATION WITHIN MY MONITOR
PROGRAM.  WITH THE EXCEPTION OF THE TEMPERATURE SENSITIVITY
WHICH WAS CURED BY OPENING THE BASEMENT HEAT REGISTERS, AND
THE TONE SENSITIVITY WHICH WAS CURED BY FINDING THE RIGHT
KNOB SETTING RANGE,  NO ERRORS IN READ OR VERIFY OPERATIONS
HAVE BEEN ENCOUNTERED.  THE 24 BIT CHECK SUM USED IN THE
MONITOR DRIVING THESE PROGRAMS IS THE PRIMARY VERIFICATION
OF THE READ OPERATIONS, WHICH IS AUGMENTED ON VERIFY OPERATIONS
BY THE BYTE BY BYTE COMPARISON OF TAPE DATA TO MEMORY DATA.


***************************************************************
THIS TEXT WAS PREPARED USING THE TEXT EDITOR PROGRAM BUILT INTO
MY MONITOR, THE FIRST ITEM WRITTEN AND DEBUGGED USING JACK
EMMERICHS' ASSEMBLER DESCRIBED (PART 1) ELSEWHERE IN THIS ISSUE
OF BYTE.       CTH 1/27/77
***************************************************************
```

```
211E 84 01    >       ANDA  #1
2120 26 00    >       BNE   U1IB
2122 7A 00 50 >       DEC   ST30
2125 27 00    >       BEQ   U1IQ
2127 3B       >       RTI
2128          >* START VERIFIES
2128 96 59    >U1IQ   LDAA  RATE
2126 01
212A 97 50    >       STAA  ST30
212C 86 08    >       LDAA  #8
212E 97 52    >       STAA  BITC
2130 CE 00 00 >       LDX   #U1I3
2133 DF 04    >       STX   NMIV
```

implemented as actual memory locations at all. Instead, the address selection outputs of IC1 for these addresses are used to control three flip flops which determine the source of the NMI interrupt generated by this circuit. The most important flip flop is formed by IC6a and IC6b. This is the interrupt mask flip flop, which is set (allowing interrupts) by a reference to hexadecimal addresses 0161 or 0162. This flip flop is cleared by either a system reset or a reference to location 0163. The system reset aspect is quite important, since it would not be too useful to have the system power up with these interrupts enabled and no interrupt handler in memory, and when developing software for the interface it helps to be able to reestablish normal interrupt free operation by pushing reset.

The flip flop formed by IC8a and IC8b is set by reference to address 0162 and is used to select the AC-30's tape derived clock as the source of interrupts. The flip flop formed by IC8c and IC8d is set by reference to address 0161, and is used to select the 4800 Hz oscillator as the source of interrupts for output use. Both of these interrupt selection flip flops are cleared by a reference to location 0163. In building this circuit for some other processor, you'll probably have a different set of IO addresses. Modification of the address selection logic (and the software of listing 1) will probably be required.

Output Latch and Input Interface

The rest of the circuit in figure 2 consists of the output latch and RS-232 level shifting buffers associated with address 0160. Writing data to location 0160 sets the value contained in the 8 bit 74100 latch, IC14. The four high order bits of this latch are used to drive the AC-30's motor control lines directly as TTL signal levels. (See table 1 for a listing of command codes corresponding to the wiring of figure 2.) The low order four bits are wired to a DS1488 line driver needed to convert TTL levels to RS-232 levels. The low order bit drives the RS-232 output to the AC-30, with the remaining three lines available but unused at present. If at some later date it becomes necessary to program a communications interface to some other RS-232 device, this hardware now lies ready and waiting in my system. The latch is written by combining the processor's RW line (inverted) with the address selection signal to generate a clock pulse.

For inputs from the external world, the bus interface gates simply listen to data lines when the address is selected. The low order four bits of an input word are defined by the

Listing 1, continued:

```
2135 3B         >       RTI
2136            >* FALSE START
2136 CE 21 0A >U1IB  LDX   #U1I1
2121 14
2139 DF 04      >       STX   NMIV
213B 3B         >       RTI
213C            >* BIT READ STATES
213C 7A 00 50 >U1I3  DEC   ST30
2131 21 3C
213F 27 00      >       BEQ   U1ID
2141 3B         >       RTI
2142 96 59      >U1ID  LDAA  RATE
2140 01
2144 97 50      >       STAA  ST30
2146 7A 00 52 >       DEC   BITC
2149 26 00      >       BNE   U1IE
214B CE 00 00 >       LDX   #U1I4
214E DF 04      >       STX   NMIV
2150 B6 01 60 >U1IE  LDAA  AC30
214A 05
2153 46         >       RORA
2154 96 53      >       LDAA  DATL
2156 46         >       RORA
2157 97 53      >       STAA  DATL
2159 3B         >       RTI
215A            >* ONE STOP BIT
215A 7A 00 50 >U1I4  DEC   ST30
214C 21 5A
215D 27 00      >       BEQ   U1IF
215F 3B         >       RTI
2160 CE 21 0A >U1IF  LDX   #U1I1
215E 01
2163 DF 04      >       STX   NMIV
2165 73 00 51 >       COM   BITV
2168 3B         >       RTI
```

```
2169 7D 00 51 >U1RD  TST   BITV
2004 21 69
216C 27 FB      >       BEQ   U1RD
216E 96 53      >       LDAA  DATL
2170 7F 00 51 >       CLR   BITV
2173 DE 15      >       LDX   MEM
2175 A7 00      >       STAA  0,X
2177 39         >       RTS
2178 7D 00 51 >U1VY  TST   BITV
200A 21 78
217B 27 FB      >       BEQ   U1VY
217D 96 53      >       LDAA  DATL
217F 7F 00 51 >       CLR   BITV
2182 DE 15      >       LDX   MEM
2184 A1 00      >       CMPA  0,X
2186 26 00      >       BNE   U1VE
2188 39         >       RTS
2189 DE 33      >U1VE  LDX   ERRS
2187 01
218B 08         >       INX
218C DF 33      >       STX   ERRS
218E 39         >       RTS
218F 7F 01 63 >U1IC  CLR   ACOF
20B2 21 8F
2192 86 7F      >       LDAA  #$7F
2194 B7 01 60 >       STAA  AC30
2197 86 FF      >       LDAA  #$FF
2199 B7 01 60 >       STAA  AC30
219C 86 FA      >       LDAA  #250
219E 37         >       PSHB
219F BD 01 38 >       JSR   MILI
21A2 33         >       PULB
21A3 BD 01 38 >       JSR   MILI
21A6 39         >       RTS
21A7            >       END
```

```
*** SYMBOLS:

AC30 0160  ACGO 0161  ACOF 0163  BITC 0052  BITV 0051
CLOS 200C  CY30 2012  CYZN 2029  CYZR 2030  DATL 0053
DATR 0054  DELA 0057  ERRS 0033  FLAG 002E  GONE 209F
HALF 005A  MEM  0015  MILI 0138  NEXB 201F  NMIV 0004
NXDA 0055  OPE1 2000  OPE2 200F  RATE 0059  READ 2003
ST30 0050  U1CL 20AC  U1I1 210A  U1I2 211B  U1I3 213C
U1I4 215A  U1IA 211A  U1IB 2136  U1IC 218F  U1ID 2142
U1IE 2150  U1IF 2160  U1IO 20EA  U1IQ 2128  U1O1 2097
U1O2 20C3  U1O3 20C7  U1O4 2093  U1OC 20B4  U1OO 206B
U1OP 205A  U1RD 2169  U1VE 2189  U1VY 2178  U1WR 2037
U2OP 2054  UUOP 205E  VRFY 2009  WRIT 2006
```

Table 1: Summary of the addresses and commands used in programming the AC-30 interface of figure 1.

Hexadecimal Address 160, symbol "AC30" in listing 1:

Input from the four RS-232 receivers of IC12 is available in the low order bits of data referenced at location 160. The low order bit, bit 0, is assigned to the AC-30 by wiring IC12 pin 13 to the pin identified as "To Computer's RS-232 Input" in the AC-30 wiring diagram. High order bits are grounded to define unused logical zero inputs.

Output from the four RS-232 drivers of IC13 is available in the low order bits of data written to location 160. The low order bit, bit 0, is assigned to the AC-30 by wiring IC13 pin 11 to the pin identified as "To Computer's RS-232 Output" in the AC-30 wiring diagram.

The high order bits are used to control the two motor control flip flops of the AC-30. Forcing the RS-232 output to remain at logical 1 (the negative RS-232 state), the following command codes are used (hexadecimal):

Code	Command	
EF =	Turn on output motor	All codes assume negative going pulses activate commands (see AC-30 documentation for settings of the options).
DF =	Turn off output motor	
BF =	Turn on read motor	
7F =	Turn off read motor	

When setting data in the low order bits, the high order data sent to location 160 should be masked by a logical sum (OR) with hexadecimal F0.

Hexadecimal Address 161, symbol "ACGO" in listing 1:

Any reference to this address turns on the 4800 Hz NMI interrupt source.

Hexadecimal Address 162, symbol "ACGO+1" in listing 1:

Any reference to this address turns on the NMI interrupt source derived from the tape data via the receiver section of the AC-30.

Hexadecimal Address 163, symbol "ACOF" in listing 1:

Any reference to this address turns off all NMI sources in this interface, as does a system reset.

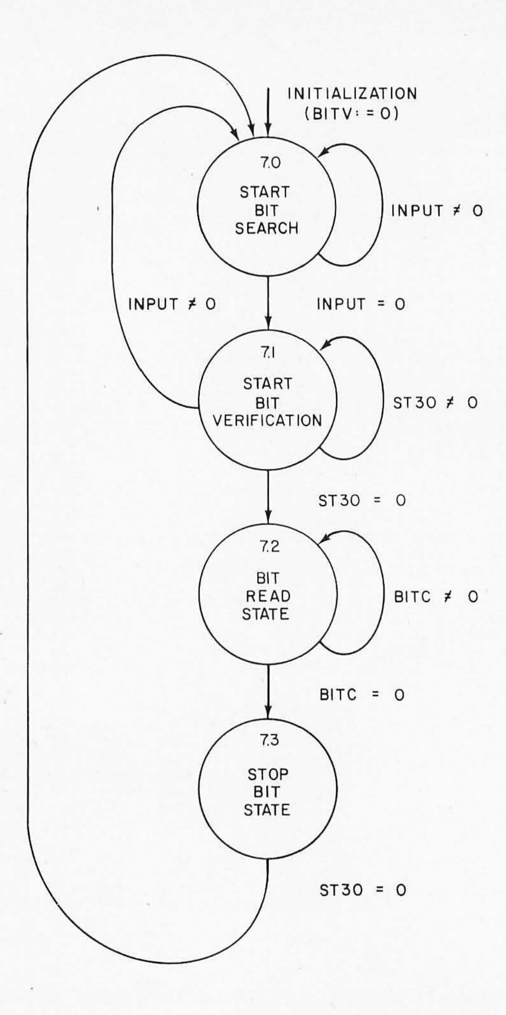


Figure 3: The finite state machine for input. The input decoding is performed by a "finite state machine" implemented in software. The state of the input decoder is set by the choice of NMI interrupt vector. This diagram shows the conditions under which transitions are made, but omits any reference to the detailed processing. See procedures 7.0 to 7.3 and listing 1 for complete details. This input state machine is initialized by the OPEN routine and it is terminated when interrupts are turned off by the CLOSE routine.

four RS-232 compatible input lines which are converted to TTL levels by the DS1489 receiver circuit, IC12. The low order bit is committed to the AC-30's RS-232 output line by wiring of the cable, leaving the other three low order bits open for later use with an RS-232 terminal as noted above. The high order input bits are presently wired to ground, but if it ever becomes necessary to add a single bit TTL level input port to the computer, these bits can be committed to that use by wiring them to a suitable external connector.

With the Hardware Defined, Software Follows Suit

The intent of this whole exercise is to provide a set of routines for writing and reading data from the AC-30 within the context of my existing monitor and text editor program, IMP 6800 ("the Interactive Manipulator Program"). This monitor provides a general IO control structure which accesses detail byte data transfer and control routines for any given IO unit. When an IO logical unit is selected by an appropriate command, the control structure references the detail IO routines by looking up their addresses in a pointer table keyed by the logical unit identification. The control structure of the monitor is not the subject of this article, but its effect is felt in the requirement of five generic IO routines which must be written to interface any given unit such as the AC-30:

OPEN: initialization of IO operations is performed by calling an OPEN procedure; the monitor data item FLAG determines whether the unit will be opened for input or output processing.

WRITE: when output is required, WRITE is called each time a byte of data is to be transferred from the address in pointer MEM to the peripheral.

READ: for input to memory, READ is called each time a byte of data is to be transferred from the peripheral to the address in pointer MEM.

VERIFY: to check the accuracy of crucial files, a form of the read operation called VERIFY compares input data to the contents of memory at the address in pointer MEM; an error count, ERRS, is incremented if necessary.

CLOSE: when IO is complete, the CLOSE routine is used to turn off the peripheral after making sure detail transmission operations are done.

The choice of these five generic IO operations for any given IO unit is a fairly typical practice when implementing the systems software of a computer. It is particularly helpful when implementing new interfaces: The control structure of my monitor is thoroughly debugged already, so when I wrote the new routines and they bombed the first time around (as often happens) I knew for sure where the problem was.

Specifying the IO Operations

The details of the IO operations for the AC-30 are shown in this article in two forms. One form is listing 1, a detailed set of code implemented for the 6800 and assembled using Jack Emmerichs' assembler program. The second form in which the IO operations are given is in a higher level "pseudo code" form as eight self contained procedures. These procedures are written from the assembly as a form of detailed commentary on the algorithm, and can be used by owners of the other processors as a starting *point* for coding new versions of the routines.

In the notation of the programs in the procedures, square brackets [] are used to enclose extended commentary and operations defined in words. Variable names and labels are the same as those used in listing 1. With a few minor exceptions the procedures

Procedure 1: OPEN routines. The OPEN routines are used to set up output or input operations, depending upon the state of FLAG, a variable which should be set to 0 for input, nonzero (example: hexadecimal FF as in listing 1) for output. Output OPEN operations consist of turning on the motor, then recording a leader (using the output interrupt of procedure 5), then initializing the interrupt handler for output operations. An input OPEN consists of turning on the input motor, then waiting a half a second before initializing the input interrupt which scans for the asynchronous data format used by this software. On leaving the OPEN routines, the software which uses this IO discipline can proceed to the details of byte by byte data transfer using WRITE, READ or VERIFY.

```
1.0      Entry U2OP [1200 bps data rate initialization];
1.1      RATE  := 4 [interrupts per bit];
1.2      Go to 1.5;
1.3      Entry U1OP [300 bps data rate initialization];
1.4      RATE  := 16 [interrupts per bit];
1.5      HALF  := RATE / 2 [set up half bit period value];
1.6      if FLAG = output value then do [an output open procedure];
1.6.0        [turn on the AC-30 output motor];
1.6.1        DELA  := 50000 [set up output write delay count for 10 seconds];
1.6.2        DAT   := $'FFFF' [initialize data output buffer to null data];
1.6.3        NXDA  := $'FFFF' [initialize next data as null data];
1.6.4        ST30  := RATE;
1.6.5        BITC  := RATE;
1.6.6        NMIV [nonmaskable interrupt vector] = [address of output open interrupt service routine, procedure 5];
1.6.7        [clear "open done" flag, the accumulator];
1.6.8        [turn on output 4800 Hz interrupt source];
1.6.9        Wait until end of leader [after DELA count is exhausted];
1.6.10       return;
1.6.11       end;
1.7      else do [an input open procedure];
1.7.1        [turn on the AC-30 input motor];
1.7.2        BITV [data ready flag]  := 0;
1.7.3        wait 0.5 seconds;
1.7.4        NMIV [nonmaskable interrupt vector] = [address of input start bit search, U1I1];
1.7.5        [turn on input ≈4800 Hz tape derived interrupt source];
1.7.6        return;
1.7.7        end;
```

Procedure 2: The WRITE routine. This routine must be preceded by an output OPEN operation which initializes the interrupt service of procedure 6. It can be called over and over again until the data to be written is exhausted. Each WRITE call transfers one byte from the location addressed by variable MEM to the output file. The software which uses WRITE is responsible for incrementing and keeping track of MEM when a block of data is being written.

```
2.0      Wait until BITC [bit count] is equal to zero;
2.1      DAT [2 byte output field]  := NXDA [next output, computed by previous call];
2.2      BITC   := 11 [1 start, 8 data, 2 stop bits minimum];
2.3      NXDA  := $'FF00' | (@MEM) [move data at MEM into new output string along with null trailing bits];
2.4      NXDA  := [arithmetic left shift of NXDA];
2.5      return;
```

Procedure 3: The READ routine. This routine must be preceded by an input OPEN operation to initialize the input interrupt service procedures. It is called over and over again until enough data has been transferred. Each READ call transfers one byte from the tape into the location addressed by variable MEM. The software which uses READ is responsible for incrementing and keeping track of the address in MEM, and for determining how long a block of data is being read. READ coordinates operation with the input interrupt service routines through the variable BITV which is used as a flag.

```
3.0      Wait until BITV = $'FF' [synchronize with flag set by interrupt service];
3.1      (@MEM)  := DAT [move input byte from buffer to address in pointer MEM];
3.2      return;
```

Procedure 4: The VERIFY routine. This routine works exactly like READ with one exception: Instead of dumping the input data into memory at location MEM, it compares the input data to the data in memory and increments the error counter ERRS if there is any difference. As with READ and WRITE, the software which calls VERIFY is responsible for incrementing MEM and keeping track of the data being checked.

```
4.0     Wait until BITV = $'FF' [synchronize with flag set by interrupt service] ;
4.1     if (@MEM) not equal to DAT then ERRS := ERRS + 1 [count the differences] ;
4.2     return;
```

Procedure 5: The output OPEN interrupt service routine. This procedure is used to count down a delay value initialized in DELA when OPEN begins its processing. OPEN ultimately terminates when this interrupt routine finds a zero value in DELA and signals the OPEN routine via the value in BITV.

```
5.0     DELA [leader interrupt count, 16 bits] := DELA - 1;
5.1     if DELA = 0 then do;
5.1.0      NMIV := [address of output data transfer interrupt service, CY30] ;
5.1.1      BITV := 1;
5.1.2      [signal end of output leader delay] ;
5.2     return from interrupt;
```

Procedure 6: Output data transfer interrupt service. This routine, which is initialized as the interrupt service routine at the end of the OPEN procedure, does the detail work of transferring data to the AC-30 from the computer. The variable ST30 is used to count down the interrupts within each output bit (see figure 1), and the variable BITC is used to count the number of bits in a character being transmitted. BITC is also used as the flag to signal to WRITE that a character is done, so its counting down is halted at zero by line 6.4. Whenever a bit is completed, the count of interrupts per bit is reinitialized in variable ST30. The variable BITC is reinitialized by WRITE when a new output word has been defined. Data is right shifted out the low order bit of a 2 byte field called DAT in this procedure, identified as DATL and DATR in listing 1.

```
6.0     [output data bit] := BITV [latest bit value] ;
6.1     ST30 [output waveform state] := ST30 - 1;
6.2     if ST30 is not zero then return from interrupt;
6.3.0   BITV [next bit value] := [low order bit of 2 byte output field DAT] ;
6.3.1   DAT := [2 byte arithmetic right shift of DAT] ;
6.4     if BITC greater than zero then BITC := BITC - 1;
6.5     ST30 := RATE [number of interrupts per bit transmitted] ;
6.6     return from interrupt;
```

follow the logic of the various sections of listing 1, but the order of the procedures differs. Listing 1 reflects the order in which the routines were actually written, while the procedures are numbered in an aesthetically nicer arrangement.

In interfacing to the monitor, the only difference between 300 bps and 1200 bps is in the definition of the number of interrupts per bit, a value which is defined in the OPEN procedure for each data rate. Thus there are two entry points in the OPEN procedure, corresponding to logical unit 1 (300 bps) and logical unit 2 (1200 bps) of my monitor. The WRITE, READ, VERIFY and CLOSE routines are shared between the two logical unit designations, although the names used only reflect unit 1.

Using Interrupts

All the IO implemented in this approach uses the interrupt from the AC-30 or the oscillator of IC9 to time operations. Thus there are three interrupt service routines specified, as procedures 5, 6 and 7. The output open interrupt service routine of procedure 5 has the simple task of counting down DELA interrupts as set by the output OPEN procedure, after which it will set up the normal output interrupt service and signal the end of the leader. The normal output interrupt service routine counts down the number of interrupts per bit, and when necessary shifts a new bit out of the output buffer, DAT, to be sent to the AC-30. It also counts bits and signals the WRITE procedure that it is ready for the next character. Figure 1 shows the progression of values of ST30 and BITC during output.

The input interrupt structure is much more complicated. There are four states to the finite state machine which handles data input, as shown in figure 3. These states are implemented by the current value of the NMI interrupt pointer, which selects one of the four input routines of procedures 7. The conditions for change of state are noted on the transitions in figure 3, but details of each state should be extracted from the procedures. The input interrupt algorithm first enters a start bit search state of procedure 7.0; when the start bit is found, it is verified in the second state of procedure 7.1 by checking to make sure that it remains low for half of a bit period (marked by counting down the value in variable HALF). When the start bit is verified, the data transfer state is entered and a total of eight bits are shifted into the input buffer DATL. After eight bits are shifted, the algorithm enters the stop bit state of procedure 7.3 for one bit period, where a flag is altered to signal that data is ready. Within 1.5 bit times of the flag change, the READ or VERIFY routine must transfer the data from DATL to its destination in memory or an alternate buffer area. This timing requirement of 1.25 ms for 1200 bps data or 5 ms for 300 bps data is due to the fact that the first bit of the next character will be shifted into DATL 1.5 bit periods after the stop bit state is executed. The time tolerance can be increased to a full character time (9.2 ms at 1200 bps) if procedure 7.3 is modified to transfer the contents of DATL into a buffering byte, for example DATR.

Some Comments on the CLOSE Routines. . .

The CLOSE routine given in this version

of the software is far from perfect. It might even be called a bit "inelegant." Any good programmer will see some rather obvious holes in the system: I really don't need four seconds worth of interrecord gap after data is written at 300 bps. Similarly, for input operations, I should have allowed the motor to remain on for a period of time equal to the null period which would have been generated by the output CLOSE operation for the same block. Given manual manipulation of tape position using the mechanical position counter, this CLOSE algorithm is adequate. But if you intend to do tape to tape editing with two cassette drives and motor control, you'll probably want to improve the CLOSE algorithm to allow more reliable electronic positioning via motor control.

Does It Work?
Yes, but 1200 bps Is Still Too Slow!

Since getting the software described here into the state seen in listing 1 and patching it into my monitor program December 26 1976, this software has been my main mass storage IO method. In a month of intense use, I have yet to drop a bit (although while testing I had to experiment to find the right cassette recorder volume level range on playback). My homebrew system has recently grown to 24 K of programmable memory with 12 K bytes of text editor string area, 4 K bytes for Jack Emmerichs' assembler program (see April and May 1977 BYTEs), 4 K bytes for assembler output strings, and 4 K bytes for my monitor and text editor programs. I have found this new 1200 bps data rate to be an indispensable time-saver compared to my older hardware method which ran at about 200 bps. Since the software described here works at 300 bps as well as at 1200 bps, I can now load and write cassettes using the physical format of the Kansas City standard. This of course finally achieves the real purpose of buying the AC-30.

With this interface I can load the assembler program in 50 seconds, and completely define the text editor string area in about 150 seconds. It has really speeded up the software development operations which are the real joy of a computer applied to personal needs and goals.

Of course, having accomplished this is not enough: My next challenge, in design as of this writing, is a 3M drive controller interfaced to a $500 surplus drive purchased from Boston's American Used Computer Corporation via their back door, The Computer Warehouse Store. When this project is done, my system will have some real IO capability in the form of an electronically controlled file system with 2 million bytes on line, 40,000 bps data transfer rate and 43 second maximum access time to any given block of data. But for the next six months or so, I anticipate frequent and reliable use of this 1200 bps modification to the AC-30's purpose in life.■

Procedure 7: Input data transfer service routines. These four procedures constitute the four states of the data input "state machine" which acts like the receiver of a typical UART or ACIA chip. The start bit search state (7.0) finds a start bit which may be verified by the start bit verification state of 7.1. If the start bit was good, eight data bits are transferred by the input bit read state of procedure 7.2. Finally, one stop bit time is marked by procedure 7.3 and the receipt of data is communicated to READ or VERIFY by changing the state of BITV. Then the start bit search resumes. See figure 3 for a state diagram of this process.

```
7.0       [start bit search interrupt service, U111]:
7.0.0        If input data line is zero then do:
7.0.0.0         ST30  := HALF [set up delay for a half bit's worth of interrupts];
7.0.0.1         NMIV  := [address of start bit verification interrupt service];
7.0.0.2         end;
7.0.1        return from interrupt;
7.1       [start bit verification interrupt service, U112]:
7.1.0        If input data line is not zero then do [false start logic];
7.1.0.0         NMIV  := [address of input start bit search, U111];
7.1.0.1         return from interrupt;
7.1.0.2         end;
7.1.1        ST30 := ST30 - 1 [count down HALF bit times];
7.1.2        if ST30 := RATE;
7.1.2.0         ST30  := RATE;
7.1.2.1         BITC  := 8 [initialize one byte's worth of bits];
7.1.2.2         NMIV  := [address of input bit read interrupt service, U113];
7.1.2.3         end;
7.1.3        return from interrupt;
7.2       [Input bit read interrupt service, U113]:
7.2.0        ST30 := ST30 - 1 [count down RATE states per bit];
7.2.1        if ST30 = 0 then do [sample bit and shift into data word];
7.2.1.0         ST30  := RATE [set up for next bit];
7.2.1.1         BITC  := BITC - 1 [count down the bits];
7.2.1.2         if BITC = 0 then NMIV := [address of stop bit interrupt service, U114];
7.2.1.3         DATL = [right rotate of DATL with carry defined by input bit value];
7.2.1.4         end;
7.2.2        return from interrupt;
7.3       [Stop bit interrupt service, U114]:
7.3.0        ST30 := ST30 - 1;
7.3.1        if ST30 = 0 then do [restart start bit search];
7.3.1.0         NMIV = [address of start bit search, U111];
7.3.1.1         BITV [data ready flag for input] = $'FF' [indicate data ready];
7.3.1.2         end;
7.3.2        return from interrupt;
```

Procedure 8: CLOSE routines. After all data operations are complete, CLOSE routines are used to turn off the motor and clean up the last details of input or output. See text for some comments about the design of this routine, which could be improved considerably.

```
8.0       if FLAG = output value then do [an output procedure];
8.0.0        Wait until BITC [bit count] is equal to zero;
8.0.1        DAT := NXDAT [transfer last character to output field];
8.0.2        [write 1200 null bits by letting output interrupts run];
8.0.3        [turn off AC-30 motor control];
8.0.4        [turn off output interrupt source];
8.0.5        [wait 0.5 seconds for motor to turn off];
8.0.6        end;
8.1       else do [an input close procedure];
8.1.0        [turn off input interrupt source];
8.1.1        [turn off AC-30 motor control];
8.1.2        [wait 0.5 seconds];
8.1.3        end;
8.2       return;
```

Artificial Intelligence:

What Is It?

Richard L Rosenbaum
61-04 Little Neck Pkwy
Little Neck NY 11362

!!!!!	!!!!!	!!!!!
!!!!!	!!!!!	!!!!!
!!!!!	!!!!!	!!!!!

a) A typical nim board set up.

!!!!!	!!!!!	!!!!!
!!!!!	!!!!!	!!!!!
!!!!!	!!!!!	!!

b) A typical first move.

Decimal	Binary
15	1111
15	1111
12	1100

c) Unsafe second move.

Decimal	Binary
15	1111
3	0011
12	1100

d) Safe second move.

Decimal	Binary
1	0001
4	0100
0	0000

e) Last move-1.

Decimal	Binary
1	0001
0	0000
0	0000

f) Winning move.

Figure 1: One version of the game of nim starts with three piles of 15 counters as shown in figure 1a. Players take turns removing any number of counters, at least one, from any one pile. The player who takes the last counter loses. In the case illustrated in figure 1b the first player took three from the last pile. Figures 1c and d illustrate some techniques used in developing an infallible play of nim. The first step is to represent the number of counters in each pile in binary. Then you must determine the number of 1s in each binary column. If the number is odd, as in figure 1c, then you are in an unsafe position. You should make your move so as to leave an even number of 1s in each column as shown in figure 1d. This is known as a safe position. A point is eventually reached, as in figure 1e, where there are only two piles left and only one of them has more than one counter. The winning strategy is to take all of the counters from the pile with more than one. This causes the other person to take the last counter, as shown in figure 1f. Using these methods, an unbeatable game of nim can be developed.

If you are a typical computer hobbyist, you have probably spent (or will spend) a fair amount of time playing games — computer games, that is. Perhaps you have written computer programs to play space war, backgammon, even chess. What techniques are used in programming computer games, particularly those requiring "intelligent" decisions by the machine? The answer to this question is in the realm of artificial intelligence.

Perhaps you are trying to program a microprocessor to recognize or interpret signals from a video camera, or would like to automatically translate Morse code to printed characters. Both of these problems concern artificial intelligence, specifically in the field of pattern recognition.

What is artificial intelligence? For that matter, what is intelligence? These questions can easily lead us into a philosophical debate, but that is not the purpose of this discussion. My interest is how artificial intelligence, frequently referred to as AI, can help us solve problems such as those just outlined. What these problems have in common is the substitution of a computer (artificial mechanism) for what are ordinarily human tasks (generally assumed to require intelligence). So for the sake of expediency, let's define artificial intelligence as the mimicking of human behavior and decision making by computer.

Today, work in artificial intelligence is divided into a number of areas. Two of them have been mentioned: game playing and pattern recognition. Others include generalized problem solving, natural language comprehension and translation, and robotics. These will be examined in order.

Game Playing Programs

It is probable that once the beginning computer hobbyist learns basic programming fundamentals, the first attempt at a big program will be an entertaining one. How

BOOKS OFF THE OLD CHIP

Introducing two NEW software books by SCELBI. Two classics rewritten for immediate use with your 6800 system . . .

SCELBI'S "6800" SOFTWARE GOURMET GUIDE & COOK BOOK

9[95] ppd.

Now you can cook up delectable programs with this mouthwatering gourmet's delight of all new "6800" software. Over 200 pages. Action-packed, information-crammed, practical "how to" facts. Using the original manufacturer's recommended mnemonics and assembly formats, with hexadecimal notations, you get the description of the "6800" instruction set. General purpose routines for multiple precision operation. Programming time delays for real time applications. Random number generators. Completely assembled relocatable floating point arithmetic program, with 23 bit signed mantissa and 7 bit signed exponent. Input/output processing for basic I/O programming. Interrupt processing. Code, numeric conversion routines. Real-time programming. Search and sort routines. And too many more finger-lickin' goodies to mention here. All for only $9.95 ppd. Order today!

SCELBI'S "GALAXY" GAME FOR THE "6800"

14[95] ppd.

Captain your own crusading starship against the logic of your "6800". Your mission: search-and-destroy a random number of alien ships. But, don't run out of time, out of fuel, out of ammunition or out of the galaxy. Your galaxy consists of 64 quadrants, subdivided into 64 sectors. Plan your mission to destroy all aliens. But, every time you move you lose a stardate and precious fuel. Don't run into a roaming star that could damage your ship! And, don't forget how much fuel your warp factor uses! Suddenly, *Condition RED! Alien in sight!* How big is he? Fire a phasor or torpedo? He's damaged or destroyed. But, you've used up valuable fuel! Does he fire back? How much fuel was used for protective shields? Be careful. You're running out of time and fuel. You get the idea. You must maneuver logically, strategically, carefully . . . to complete your mission. Here's the multidimensional computer game you've asked for. Using the original manufacturer's recommended mnemonics and assembly format, with hexadecimal notations, you've got a total book form program in machine language, for 4K memory, with flow charts, illustrations and more. All this for $14.95 ppd. Order today!

BOTH BOOKS ARE AVAILABLE NOW! ORDER YOUR COPIES TODAY!



1322 Rear Boston Post Road, Milford, CT 06460 • Telephone: 203/874-1573

Prices shown for North American customers. Master Charge, Postal and Bank Money Orders preferred. Personal checks delay shipping up to 4 weeks. Pricing, specifications and availability subject to change without notice.

many of us, once we have BASIC up and running, quickly type in our copy of Star Trek? We all have some interest in computer games.

While many games use artificial intelligence concepts in their programming, not all of them do. For instance, Star Trek usually consists of a program that keeps track of time, energy levels, scores and other parameters. The only decision to be made is generally, "Which way do I fire?", which only requires use of a simple formula. Of course, there are exceptions where some super program simulates the alien cunning of a Klingon commander.

Artificial intelligence techniques are required, however, for the automation of most traditional multiple player games. These range from the simple binary strategy of nim (see figure 1) to the complexities of chess. The case of nim illustrates one extreme of an intelligent program: the perfect player. The game is simple enough to develop a fast technique to determine the best move. Such is not the case in chess. The important point here is that both chess and nim are deterministic; ie: if you have enough time, you can go through every possibility that can arise. However, with games as complex as chess, a program would take a tremendous amount of time to check all the possible moves. This is where artificial intelligence techniques come to the rescue.

Let us examine how a human player makes decisions in a game such as chess. This technique, saying, "How would I do it?", is invaluable in artificial intelligence work. Obviously, we do not go through every possibility. What we do instead is to make trial moves and evaluate the board situation at that point. If the situation is too dangerous, we discard that possibility and check others until we make a decision.

So we need an evaluation function. The quality of this function will strongly influence how good a game the computer can play. In 1967, A Samuel wrote a program to play checkers. In it, he used an evaluation formula that took into account such parameters as number of pieces, number of kings, how far the kings were advanced, etc. Samuel, however, did not know how to weight the various items to determine their relative importance. So he made a set of trial values and incorporated into the program a mechanism that changed these values if it was losing, in order to try to find an optimum strategy. In this way, he wrote a program which "learned" to play better checkers!

Of course there is more to playing a game than just evaluating positions. We must generate trial moves in some orderly fashion. The most efficient way to do this is to make a tree of possible moves as shown in figure 2 for example. This figure is called a tree because of the way it resembles an upside down tree with branches.

As can be seen, trees can rapidly expand as they grow downward. The tree shown in figure 2 has 1.2×10^{56} branches. With the limited memory space of a typical microprocessor, we must find some way to minimize its size. One way is to copy the human player, and not expand any branches that do not look promising, ie: do not have a high evaluation. Other aspects of artificial intelligence concern optimization of the evaluation, speeding up tree generation, and minimizing other memory requirements. Often,

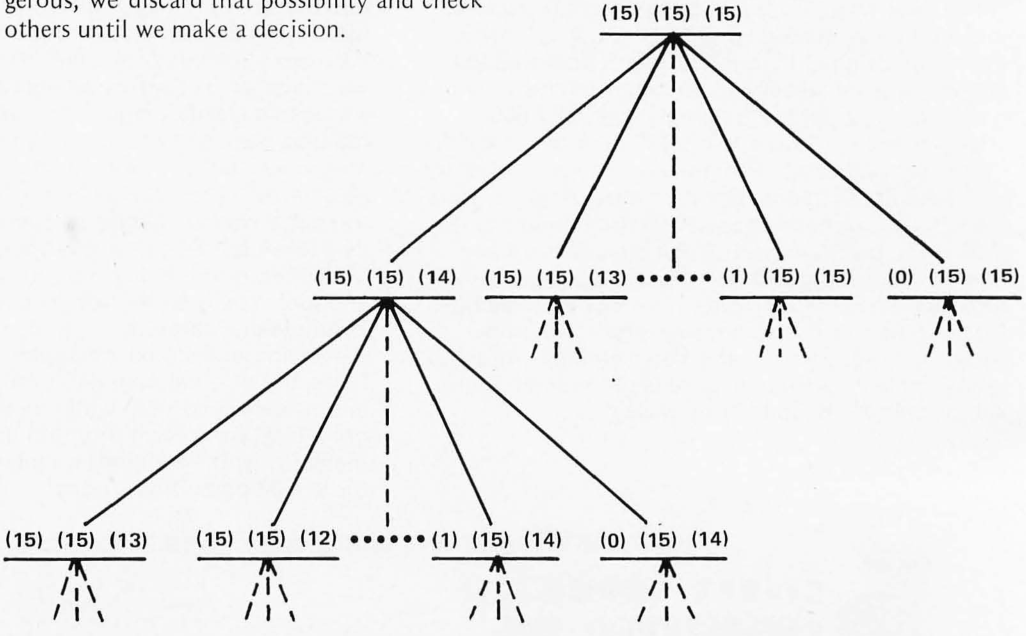


Figure 2: This is the beginning of a game tree for the game of nim as discussed in figure 1. Only a few of the branches are shown for the sake of clarity. Using this version of the game there are 45 different ways to play the game!

Subsystem "B" makes the computer you already have work almost as well as a new Sol-20.

Add it now, and for a limited time we'll throw in a new module with 2048 bytes of preprogrammed PROM or ROM and 1024 bytes of RAM memory free!

Processor Technology's Subsystem "B" puts together the major system elements you need to get your Altair or IMSAI up and running.

You get both RAM and PROM memory, parallel, serial, cassette and video display interfaces, and software. Software includes a bootstrap loader program so you can load any Processor Technology CUTS ("Byte/Kansas City") 1200 BAUD cassette tapes. Standardized subroutines in ROM, similar to those in a Sol personality module, are used by many Processor Technology software packages to improve program efficiency. You'll find you rarely need to touch your front panel switches. With our Subsystem "B" you are up and running as soon as you turn on the power.

Three subsystems are available, depending on your memory requirements. Each Subsystem "B" includes five S-100 bus compatible modules as listed below. Each is dependent upon our new GPM module (GPM = General Purpose Memory) which provides 1024 bytes of low power static RAM and 2048 bytes of preprogrammed ROM or EPROM as well as space for up to 8192 bytes more of ROM or 2708 type EPROM. The GPM module accepts the new ROM version of our well known ALS-8 Editor/Assembler software package.

With Subsystem "B" you have guaranteed compatibility with all Processor Technology software and hardware products.

Subsystem Model	B70	B110	B190
Total memory provided (bytes)	7168	11264	19456
Display I/O	VDM-1	VDM-1	VDM-1
Parallel, Serial I/O	3P+S	3P+S	3P+S
Tape Cassette I/O	CUTS	CUTS	CUTS
Memory	**4KRA**	**8KRA**	**16KRA**
Price	$594	$730	$964

NOTE: The GPM module with both RAM and PROM is included FREE in all the above subsystems. The GPM kit is $129 if purchased separately.

The ALS-8/ROM chip set is $159. SIM-1 & TXT-2/ROM add-on set is $60 and requires both GPM and ALS-8/ROM.

I want Subsystem "B" to get my Altair or IMSAI up and running.

☐ Enclosed is a check for $_____
California residents add 6% sales tax.
No shipping charge.

☐ Mastercharge # __________
(include Interbank No. and expiration date)

☐ Send more information

I want:
☐ B70 at $594
☐ B110 at $730
☐ B190 at $964

Name ____________________

Address ____________________

City __________ State _____ Zip _____

Processor Technology Corporation
6200 Hollis Street, Box B
Emeryville, CA 94608 (415) 652-8080

Figure 3: A syntactic analysis of the sentence, "This game is played by two people on the computer." This particular parse of the sentence, resolving the sentence into its components, means that two people are engaged in playing a game that somehow involves the use of the computer (the adverb use of the phrase "on the computer" to modify "is played").

one only stores a small local region of the tree at any one time, using rules to generate additional segments.

Pattern Recognition

The influence of computer aided pattern recognition is affecting us more each day. Our checks have funny digits printed on them, grocery items have strange stripes; you also might have run into forms that required you to write characters exactly as indicated. A portion of artificial intelligence concerns itself with these tasks.

Pattern recognition is generally a problem of classification. We have either a single item which we wish to describe as having certain characteristics, or we have a group of objects which we wish to break into subgroups. The principal problem is determining if a particular object is in a certain class of objects. Hence, the field extends beyond merely recognizing characters on a piece of paper. Potentially, artificial intelligence techniques could help classify medical problems based upon symptoms, the field of computer aided diagnosis long found in science fiction literature.

Unfortunately, much of the work in this field involves higher mathematics which are beyond the scope of this article. However, some elementary principles may help you conduct some investigations, perhaps in computer analysis of video signals. A fundamental technique is to break your recognition problem into the simplest subproblems possible. For instance, consider the problem of analyzing faces. One would have to generate parameters for subclassifications such as color of eyes, hair, etc. In this way, each face is assigned a set of basic values, a prerequisite of any attempt to make comparisons.

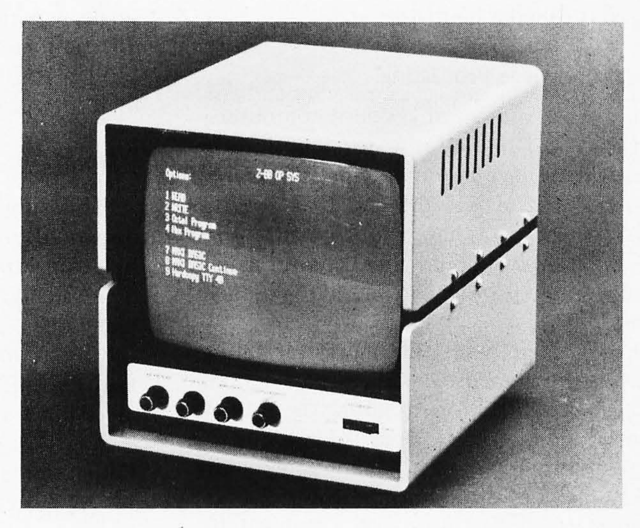
Problem Solving

Problem solving may appear to cover everything under the sun but within artificial intelligence it refers to a specific set of tasks. In artificial intelligence, a problem is said to exist when we are in an initial state, desire to reach a goal state, and have a limited set of operations that can be used to get there. Game playing is actually a subset of this, in that the initial state is the present position, and the goal state is a win. Another example is theorem proving, in which the initial state is given (remember your geometry?) and the operations are the various axioms.

There have been many attempts to develop general methods for these types of problems. In fact, an early one was called GPS, General Problem Solver. The authors of this program, A Newell, H Simon, and J Shaw, developed many of the techniques in this field. The simplest one was a brute force method, similar to our discussion of checking every possible move in a chess game. They called this the "British Museum algorithm" in reference to the fact that a monkey placed in front of a typewriter will eventually write all the books in the British Museum. So much for that idea!

Of all the work done, the most popular idea was embodied in a program called the Fortran Deductive System. This program would take the initial and goal states, and develop a set of differences. It then would methodically apply the various operations in an attempt to reach the goal. The difference it had from the brute force approach was that it knew which operations affected

The Digital Group adds character(s).



64, to be exact.

The Digital Group's computer systems have a lot of character already. Just one quick look at any of our products in their unique custom cabinets confirms that. But we believe it never hurts to add a bit more.

So, the Digital Group has added character in a big way to give an added dimension to the operation of our video-based computer systems. We are pleased to announce our new TV readout with a 64-character line. It will give your system a great deal more capability. Give it more character, if you will.

Here are the specifics on the Digital Group TV Readout and Audio Cassette Interface:

1024 Character TV Readout

- 64 characters horizontal by 16 lines
- 7x9 character matrix (effectively 7x12 due to character shifting)
- 1K on-board RAM for buffer storage—requires no main memory—completely independent
- 128 character ASCII
  - Upper case alpha
  - Lower case alpha with base line extenders (g, j, p, y)
  - Numbers and extended math symbols
  - Greek alphabet
- Software driven cursor—forward and backward
- Compatible with most microprocessors; Interfaces with 1 8-bit parallel output port
- Timebase may be driven with an external timebase (may be synchronized to TV camera, TV set, etc.)
- Readout timebase available at connector (can be used for graphic driver, etc.)
- White characters on black, and/or black on white; software selectable
- Plugs into standard dual 22-pin TVC connector on Digital Group Systems

Improved Audio Cassette Interface:

- Reliable FSK recording technique
- Uses standard unmodified audio cassette recorder
- Write cassette system uses a digitally synthesized frequency shift system, derived from TV system's master crystal oscillator
- Read cassette system easily aligned using the write system as an alignment aid.
- Runs at 1100 baud (100 characters/second)—loads 16K in 3 minutes

512 TVC to 1024 TVC Upgrade Kit:

As always, when the Digital Group extends the capabilities of our systems, it doesn't mean obsolescence for any products. We are offering an upgrade kit for present Digital Group system owners who wish to go to the longer line length. This kit uses most of the IC's from our TVC-F readout. No unsoldering is required; all new sockets, capacitors, resistors, PC board and other necessary parts are supplied.

Prices:

TVC-64—Full 64-character TV Readout & Audio Cassette Interface:
Kit — $140 Assembled — $205

TVC-64UPG—Upgrade kit from TVC-F:
Kit — $65

If you already own a Digital Group system, our 64-character line will definitely enhance its operation. If you're just looking, you might want to keep in mind that the Digital Group has a lot of characters.

Write or call now for details on our new 64-character TV readout and all our other exciting products.

the digital group INC

box 6528 denver, colorado 80206 (303) 777-7133

which differences. By the way, any technique that "intelligently" reduces a problem in complexity so that computation time is reduced is called a heuristic.

Natural Language Processing

In the early days of modern computers, the early 50s, one of the great optimistic hopes was to have automatic machine translation of text. It was thought that all you would have to do is have enough memory to store a dictionary. Today there are not many knowledgeable people who would make such a boast.

The problem with writing a translating system is the immense complexity of natural language. English, for example, is full of potential ambiguities. Does "Time flies!" mean a comment about how time is going by quickly, or does it mean a command that we should start chasing flies with a stopwatch? One can think of hundreds of such sentences.

A related problem is natural language comprehension. Wouldn't it be nice if we could converse with our processors in English instead of BASIC or assembly language? Unfortunately, the same problems are faced in this task. One way to alleviate the problem is to limit our speaking vocabulary. Monitors often do this to an extreme, limiting us to specifics such as examine and deposit. The most successful approach for larger systems involves syntax directed methods. This means that each sentence is broken down until its meaning is understood. This method has a "hopefully it will work" attitude about it. Figures 3 and 4 illustrate this technique, showing how a single sentence can be broken down into different meanings. These methods are also applied to the design of computer languages, their interpreters, and compilers.

Robotics

This subject is last for good reason. The design of robots is largely based upon the preceding topics. For vision, if it is capable of it, a robot would use pattern recognition techniques. For performing tasks, problem solving is involved. Robotics is an application of artificial intelligence to hardware of mechanical systems as a control element.

Conclusion

Artificial intelligence probably offers something for every programmer . . . certainly if you like to play games. But there are other aspects that are just briefly touched upon here. If we are building a large library of information, perhaps we would like to use an English-like inquiry system. The computer directed medical diagnosis could be extended to auto maintenance, or even computer repair. Whenever you want some of your thinking done for you, automate it with artificial intelligence.■

REFERENCES

1. Gelerntner, H, "Realization of a Geometry Proving Machine," *Computers and Thought*, McGraw-Hill, NY 1963.

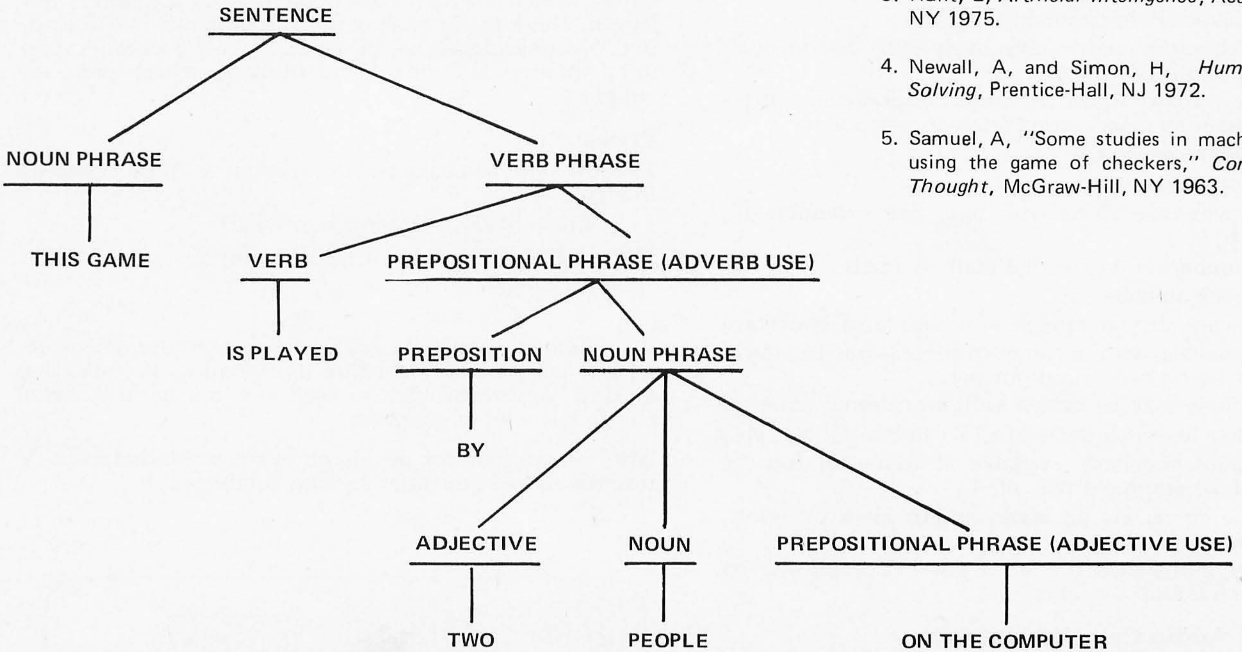
2. Gries, D, *Compiler Construction for Digital Computers*, Wiley, NY 1971.

3. Hunt, E, *Artificial Intelligence*, Academic Press, NY 1975.

4. Newall, A, and Simon, H, *Human Problem Solving*, Prentice-Hall, NJ 1972.

5. Samuel, A, "Some studies in machine learning using the game of checkers," *Computers and Thought*, McGraw-Hill, NY 1963.

Figure 4: Another parse of the sentence of figure 3. This particular parse is evaluated as meaning that two people are on top of the computer (adjective use of "on the computer") and are engaged in playing a game. Both the parse of figure 3 and this one are valid and demonstrate a major difficulty that is encountered when trying to use computer evaluation of English or other natural languages.



BASIC

The handle for the tool.

A microcomputer without software is a tool without a handle. PolyMorphic systems BASIC is the handle on the POLY 88; this provides the interface between user and computer. Our BASIC fits the POLY 88 like a finely balanced handle fits a quality tool. PolyMorphic Systems Basic is an extremely efficient way to program. It makes possible the immediate use of the POLY 88 for a wide range of engineering, scientific and general problem solving. In conjunction with either System 7 or System 16 PolyMorphic Systems BASIC is the reliable and ready to go microcomputer tool on the market. No more waiting for the long-promised software system. Among our best BASIC features: graphic plotting function, tape save and dump with named files, time function, and self-explanatory error messages. We believe the best tool — the POLY 88 — must have the best handle: PolyMorphic Systems BASIC.

Polymorphic Systems 11K BASIC

Size: 11K bytes.
Scientific Functions: Sine, cosine, log, exponential, square root, random number, x to the y power.
Formatted Output • Multi-line Function Definition • String Manipulation and String Functions • Real-Time Clock • Point-Plotting on Video Display • Array dimensions limited by memory • Cassette Save and Load of Named Programs • Multiple Statements per Line • Renumber • Memory Load and Store • 8080 Input and Output • If Then Else • Input type — ahead
Commands: RUN, LIST, SCR, CLEAR, REN, CONT
Statements: LET, IF, THEN, ELSE, FOR, NEXT, GOTO, ON, EXIT, STOP, END, REM, READ, DATA, RESTORE, INPUT, GOSUB, RETURN, PRINT, FILL, OUT.
Built in Functions: FREE, ABS, SGN, INT, LEN, CHR$, VAL, STR$, ASC, SIN, COS, RND, LOG, TIME, WAIT, EXP, SORT, CALL, EXAM, INP, PLOT.

Systems Available. The POLY 88 is available in either the kit or assembled form. It is suggested that kits be attempted only by persons familiar with digital circuitry. The following is a list of the systems available.

System 1: is a kit and consists of the Poly 88 chassis, CPU and video circuit cards only. Requires keyboard and TV monitor for operation. **$595.**

System 2: Consists of System 1 with the addition of the cassette interface circuitry-requires cassette recorder. **$690.**

System 3: System 2 with 8K memory card, BASIC and assembler cassette tapes. **$990.**

System 4: Also a kit, but containing in addition to System 3 a keyboard, cassette deck and TV monitor. **$1350.**

System 7: Consists of an assembled and tested POLY 88 with 8K of memory, keyboard, TV monitor, cassette recorder, 8K BASIC and Assembler cassette tapes. **$1750.**

System 16: Consists of an assembled and tested POLY 88 with 16K of memory, keyboard, TV monitor, cassette recorder, 11K BASIC and Assembler cassette tapes. **$1995.**

460 Ward Dr Santa Barbara. CA 93111 (805) 967-2351

PolyMorphic Systems



Have you written Software for your Altair™ Computer?

The Altair 8800 computer was the first micro produced for the general public and remains number one in sales, with more than 8,000 mainframes in the field. The wide acceptance of the Altair computer and its rapid adaptation to many diversified applications has truly turned the dream of the affordable computer into a reality.

Yet the machine itself, remarkable as it is, represents only the beginning. The right Software, tailored to meet a user's specific requirements, is a vital part of any computer system. MITS wants to insure that Altair users everywhere have the best applications software available today and in the future. For this reason, a new MITS subsidiary, the ALTAIR SOFTWARE DISTRIBUTION COMPANY, has been formed. Its purpose: to acquire the highest quality software possible and distribute it nationally through Altair Computer Centers.

That's where you come in. The ASDC will pay substantial royalties to the originators of all software accepted into the ASDC library. If you have written business, industrial or commercial use software for the Altair 8800, ASDC wants to hear from you. It is the aim of the ASDC to stimulate and reward creativity in producing useful software that makes those dreams of "computers for everyone" come true. The ASDC will select only software that measures up to its high standards for system design, coding and documentation. The software will then be further documented and distributed through Altair Computer Centers around the country.

For more information on how to submit software to the ASDC, ask your Local Altair Computer Center for an ASDC Software Submittal Packet or contact the ALTAIR SOFTWARE DISTRIBUTION COMPANY.

A subsidiary of MITS

ASDC

ALTAIR SOFTWARE DISTRIBUTION COMPANY

3330 Peachtree Road, Suite 343 Atlanta, Georgia 30326 404-231-2308

see next page for a listing of Altair Computer Centers

Ask BYTE

MORE ROBOT REACTIONS

In reading your January 1977 issue I noticed a letter from Dean Blondefield *[page 140]*, asking about books about building your own robot.

In the December 1976 issue of *Scientific American* (in the "Book Review" section) a review may be found of a book entitled *Build Your Own Working Robot* by David L Heiserman, Tab Books, Blue Ridge Summit PA.

From the description in the review this would appear to satisfy at least some of Mr Blondefield's interest.

R S Schlaifer
1500 Rollin
South Pasadena CA 91030■

ROBOT LITERATURE RESOURCES

I would like to pass along to Dean Blondefield of your January 1977 "Ask BYTE" column, page 140, and any other interested BYTE readers, the few "bytes" of information I have concerning robots.

There have indeed been too few articles and books published about my favorite subject of robots, as the editor claims. A few of the few were: *Popular Electronics*, December 1958, January 1959, and *Popular Electronics*, March 1962. The robots described by these articles were very primitive in nature but would be useful to someone starting out because they clearly demonstrate simple cause and effect situations, and provide a mechanical test bed for further experiments. A series of manuals on home construction of robots was published by a company called Electric Brain Enterprises, in Utah, about 1960. The first of these manuals, *Robots I*, is dedicated largely to the mechanical problems which the home builder may encounter. I never received the additional manuals in the series so I have no idea what they contained in the way of information.

The latest work to come across my desk is *Build Your Own Working Robot*, by David L Heiserman, published by Tab Books, Blue Ridge Summit PA 17214. This book of 234 pages published in April 1976 is by far the best I have ever read on the subject. I can recommend it to anyone with a clear conscience.

I firmly believe that in less than 20 years, for about the same price as a middle sized car, you will be able to purchase a machine to do most of your housework. In less than ten years you will be able to purchase, for about $1,500 of today's dollars, a primitive robot lawnmower.

To Mr Blondefield, and to anyone else who has ambitions to build the Homo Robotus, I would only give this small bit of advice: Man evolved on this planet one step at a time: Evolve your robot the same way. First model a simple lower order creature, then perhaps a robot mouse or dog. Divide the big project into many small ones, each with its own well defined goal so that you can watch your creation evolve and grow as you add each new ability or function. Man did not reach his present level in one jump or one day — neither will the Homo Robotus.

To the staff of BYTE: Thank you for an excellent publication.

James F Ward
3121 Chestnut Rd
Carrollton TX 75006

Thanks for an excellent letter. The Tab book on robots, Build Your Own Working Robot *by David L Heiserman, is good as far as it goes and is an excellent source of ideas. It is just the starting point, however, and antedates (very slightly) the advent of microprocessors as flexible control elements for the practical experimenter.* [It is available through our book service, BITS Inc.]■

MORE ON ROBOTS ... FROM THE "UNITED STATES ROBOTICS SOCIETY"

We were much interested to see the letter in the January 1977 BYTE *[page 140]*, from Mr Blondefield in Hayward, seeking a robotics organization more concerned with individual effort that with institutional and "industrial" robots.

As you will realize, we are engaged chiefly in a search, at the moment, for methods, hardware and experience that will clarify the status of robotics. We are persuaded from some years of work in closely related fields that the keys to robotics — to development of machines that have behavioral characteristics like those of animals, making many of the same judgements in the same ways — have already been discovered and that it is only necessary to apply a diversity of ideas in a single working system to create a "robot" of considerable capability.

We don't expect overnight success in the search. Probably some decades of activity are ahead of us before robots of true science fiction capability are available, but for the time being we'd be dazzled by an artificial system with the behavioral characteristics and apparent intelligence of an ant — never mind an approach to human behavior. It's a long hard road.

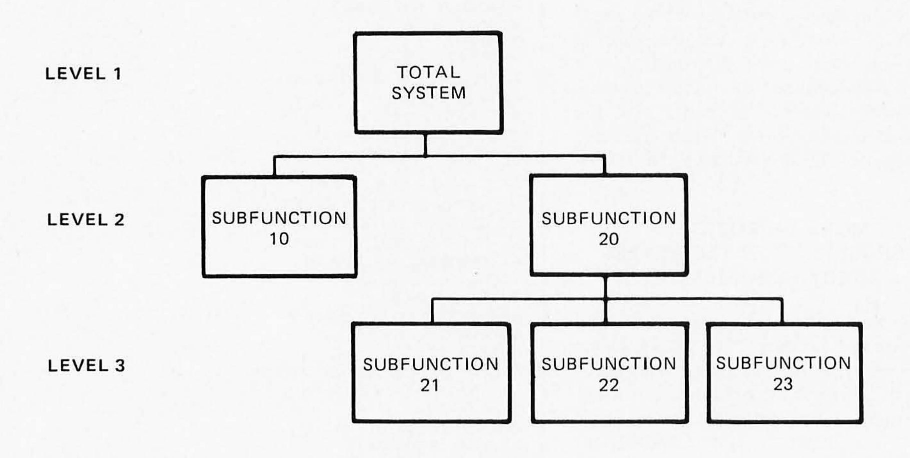
When the United States Robotics Society was formed last summer, we had no idea whether or not as many as a dozen people would be interested enough to participate, but we have taken the necessary steps to become a formal nonprofit corporation, set up carefully by lawyers to achieve a Federal tax exemption when we have the resources to pursue it.

Here's a point, by the way, on which we take issue with your reply to Mr

Continued on page 149

Designing the

Jack Emmerichs
8465 N 51st St
Brown Deer WI 53223



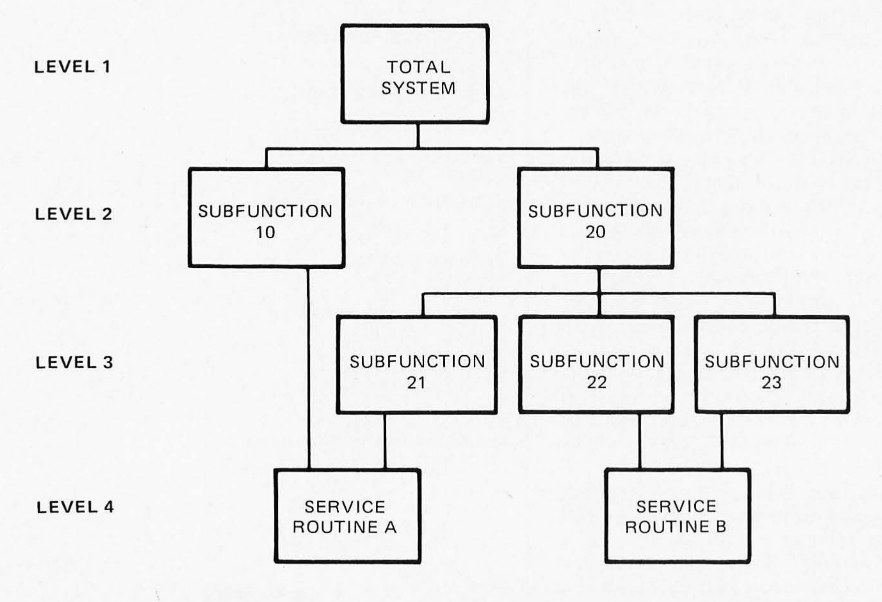


Figure 1: The top chart shows the structure of a single hierarchy diagram. The functions at each level are broken down into subfunctions at the next lower level. The relationships are shown as a simple tree structure. The bottom chart shows the structure of a network diagram. Here routines may be common to more than one high level function, and the structure of relationships can become much more complex. General subroutines that are common to many functions may be shown as a separate structure.

When I first became aware of the small systems industry, I was particularly interested in finding out what software had been developed for personal computers, especially software that would run on a minimal system configuration without much optional hardware. One of the most useful software products for such small systems is an assembler to relieve the programmer of the tedious job of programming in machine language. I found no assemblers, however, that would run on a small machine configuration.

Most assemblers required at least 8 K bytes of memory (which was beyond my initial budget projections). As two pass assemblers, many of the existing products also required that source programs be entered twice so that all symbols in the source code could be resolved. This may be difficult or cumbersome if a system does not have IO devices that can easily handle high volumes of data. Furthermore, none of the assemblers that I found could be used interactively because of this two pass design. For a minimum system without offline file storage, this can be a major problem. In many cases several IO interfaces were required to handle the input source code, printed listing, and the generated object code (machine language). Finally, most of the assemblers then available provided all the bells and whistles available in much larger programs designed to run on large timesharing systems. These large assemblers (called cross assemblers) are much more complex than I felt a small system assembler needs to be. I therefore decided to write a small but powerful assembler that would run on what to me was an affordable machine. This article is a description of the design and construction of such an assembler using structured programming techniques.

The first thing to do in any development project (though it is unfortunately often skipped in small projects) is to define the program's specifications. For this project the requirement was to develop a memory resident M6800 assembler which would:

- Assemble source code written in a free format subset of the M6800 assembly language as described in the Motorola

"Tiny Assembler"

Defining the Problem

publication *M6800 Microprocessor Programming Manual.*

- Operate completely within the first 4 K of memory including all tables, buffers, and other memory requirements.
- Completely assemble source programs in one pass to minimize IO operations.
- Require no more than one IO interface using the Motorola MIKBUG monitor.
- Support interactive operation.
- Facilitate modification and customization through the use of structured code.
- Have sufficient capacity to be able to assemble itself so that no other software would be required to generate modified versions of the assembler program.

After the design criteria were thus set down, and before the program coding was started, it was necessary to define the functions that would be needed to meet the above specifications. Functions were defined from the highest level down to the lowest, and were related to each other through hierarchy and network diagrams (see figure 1). The first series of functions that we shall consider here are those that are common to all assemblers. These functions are discussed in more detail in the article "Jack and the Machine Talk" by Robert Grappel and Jack Hemenway, published in the August 1976 BYTE, page 52.

Figure 2 shows how the network diagram worked out for an initial conception of the assembler. At the top level the assembler is shown as a single function, level 1. Below this on level 2, the first major function is usually a table initialization and housekeeping routine (START) that need not be closely examined here. The second major function is to parse and process each line of incoming code with its labels, mnemonic instructions, operands, and comments (PARSE). The final major function is usually a termination and reporting function which again need not be closely examined here (CLEANUP).

At the next lower level, level 3, several functions must be defined which will be called from the parsing routine. The first of these is a function to convert English-like mnemonic instructions into machine language operation codes (opcodes). This is complicated in the M6800 by having the opcodes for some instructions vary with the accumulator or type of addressing used. This information is developed as each line of input is processed, so the parse and translate functions must work together to develop the correct opcode. The second function required at level 3 is to process assembler directives that control the assembly process from the input stream. These may include such operations as reserving sections of memory, setting up constants, assigning values to symbols, changing the program counter, and whatever else may be defined in the language. The third function required at this level is the maintenance of a table of user defined symbols with their associated values. Entries may be added to the table, or retrieved from it as the parsing routine directs. The final function at this level is an

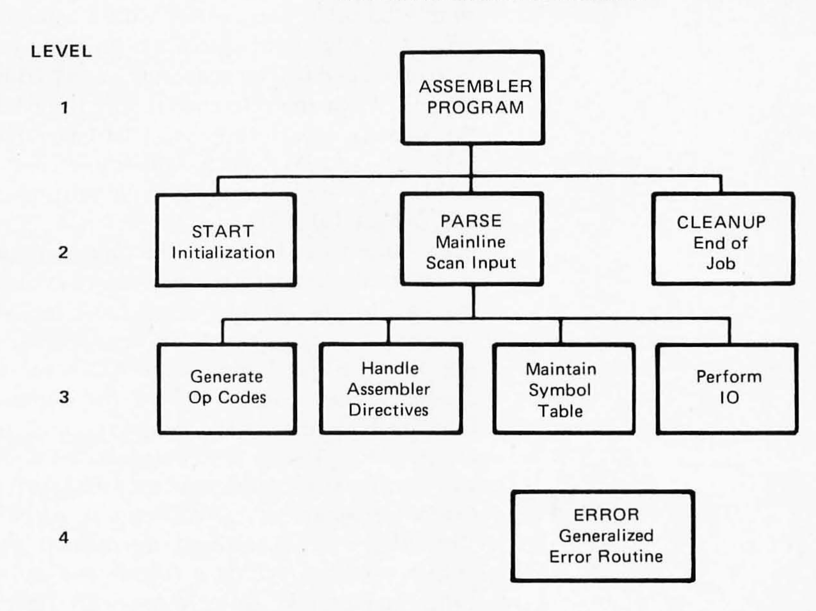


Figure 2: This is a simple network diagram showing the primary functions required by assembler programs in general. The error routine is common to all other functions.

Editor's Note:

At the present time (January 1977) Jack Emmerichs' assembler, described here and in the second part which follows next month, is my principal means of assembling programs for my homebrew 6800 based system. I've very successfully used it to assemble its own patches to fit my system, a text editor which is now part of my old hand assembled monitor, an extensive music text editor program, and two different versions (so far) of a multiprocessing music interpreter program to drive my synthesizer peripherals. After having suffered systems software withdrawal pains for over a year, it is great to get my "fix" of this intoxicating elixir. . . . CH

IO routine which receives source code from an input device, and writes listings and generated code to output devices.

The last level in this structure is shown here as a generalized error handling routine that flags and describes any ambiguities or errors in the source code. This can be described as a general service routine, and may be invoked from any function detecting an error.

Several aspects of this project required the addition of new functions or modifications to existing functions within this structure, and will be treated here in some detail. The most severe functional changes were the result of requiring the source program to be processed completely in one pass.

The problem of making the assembler a one pass type in general is: What is to be done when the incoming code references a symbol that is not yet in the symbol table but may be defined later on? This is known as a forward reference and is quite likely to occur in almost all programs of nontrivial length.

The solution to this problem will require coordination between the assembler program and a loader program. The loader is run as a separate operation that converts the object code developed by the assembler into an executable program in memory. In this case, the assembler will generate one, two or three bytes of object code for each line of source code. It will also generate the address in memory where these bytes are to be loaded. When a forward reference is made, "dummy" code will be generated instead of the normal object code. This dummy code is simply an address or offset with a value of zero. The address of where the final address is to be loaded will be generated and saved in a table. When the reference is later resolved, the correct object code must be generated with an address which will cause it to overlay the original dummy code putting in the correct value. If several references have been made to a symbol by the time its value is resolved, several sets of addresses and patches to the dummy code must be developed. The opcode which is developed and put out with the dummy code will never be changed. The operand will be the dummy value to be patched. The loader required to properly handle this type of generated code will be shown in detail next month. *[Such a loader program is effectively a second "assembly" pass; however, by making the output routine include a loader and using extra memory as we've done with Jack's assembler, the entire process can be carried out within memory nearly instantly . . .* CH*]*

Before considering the functions needed to achieve the generation of dummy code and later patches, two specific problems must be considered: What kind of opcodes and addressing formats must be handled, and how are expressions containing a forward reference handled?

First, consider the relationship between opcodes and addressing modes. For the M6800, opcodes are one byte in length, and may vary with the accumulator or mode of addressing being used. The operand may be a one or two byte address value which is a direct reference to a location in the first 256 positions of memory (zero through 255), an extended reference to any position in memory, an offset from the index register, or an offset relative to the address of the next instruction. It may also be one or two bytes of data immediately following the opcode. As the incoming instruction containing the forward reference is being processed, the addressing mode being used is usually well defined and indicates a unique opcode and instruction length. This is not the case, however, when the instruction making a forward reference is not relative, immediate, or indexed. It can then be either direct (one byte) or extended (two bytes). The difference depends entirely upon whether the final resolved address is greater than 255 or not, and at this point the address value is not known. Therefore, I established the following convention: When no addressing mode is specified in a forward reference, extended addressing will be assumed since it can refer to any position in memory. Some efficiency may be lost by not allowing the shorter form of addressing here, but most forward references are not to locations in the first 256 locations anyway, so the restriction is not a severe one. *[Using an equate (EQU pseudo operation) to define a page zero address ahead of its first use neatly solves the problem when using the assembler . . .* CH*]*

The second problem is the handling of arithmetic expressions containing a forward reference. It will require a solution built upon the logic developed so far, and will require additional coordination between the assembler and the loader. If the loader combines overlaying code with whatever is already in memory instead of using it as a replacement, the dummy code can be used to contain part of the information needed to resolve such an expression. This requires that the loader clears memory before starting the loading process and can correctly combine all single and double byte values. An expression with a forward reference is best illustrated by an example.

To load accumulator A with the address

of five less than the sum of Y and Z (both unknown at this point) in table 1, the following code could be used:

EXAMPLE1 LDAA TABLE1+Y+Z–5

The expression (TABLE1+Y+Z–5) is evaluated as far as it can be, and the value of (TABLE1–5) is put out as the dummy code. When Y and Z are resolved (in any order), they are each put out as a correcting value. The loader then combines all three values (TABLE1–5, Y and Z) to arrive at the final value. If the loader simply adds each correction to what is currently in memory, a certain amount of caution is required when using the minus sign (–) or symbols with a negative value. A further consideration is that the assembler can only check the range of relative or indexed offsets for each occurrence of dummy or correcting code. When these are combined by the loader, they may exceed allowable offset ranges. It is up to the programmer to see that they do not exceed the limits. Since the requirements for handling forward references have been examined, and methods of achieving these requirements have been developed, the first new functions required by this assembler can now be considered.

The first major change is a new functional requirement. A routine is needed to record all pending forward references in a table. The table must contain a pointer to the symbol that was referenced, the address in the source program where the reference occurred and dummy code was generated, and the type of reference that was made (one byte or two, absolute or relative). Each time the routine is invoked, a new entry is added to the table. If the table is full, an error condition is raised. The table size was arbitrarily set at 25, which has proven to be more than sufficient if the source program is properly organized. *[In writing a new version of my monitor and text editor program (about 1500 source statements) with Jack's assembler this has proved to be* the case . . . CH*]* The Set Forward Reference routine has the logical structure shown in listing 1. Note that in this structured programming notation there is only one logical entrance and one logical exit from each function. The logical exit (return) may have several physical locations, however, to reduce the amount of code used. *[The use of structured pseudocode such as that shown in listing 1 for program design was discussed in the article "Programming for the Beginner" by Ronald Herman published in the June* 1976 BYTE, *page 22.]*

The second major change is another new functional requirement. A routine is needed to generate the correcting object code when the value of an item in the new table is resolved. This routine must search the entire forward reference table for pointers to the newly resolved symbol and generate a correcting reference for each. The address at which the dummy code was generated is retrieved from the table and used as the location for the correcting reference. If the previous reference was a relative instruction, the difference between the previous address and the resolved address is calculated and a single byte is put out. If the previous reference required a one byte address (indexed or immediate one byte), one byte is generated. If it required a two byte address (extended or immediate two byte), two are generated. As each correcting item is complete, its position in the forward reference

Set Forward Reference

```
(start at beginning of forward reference table)
DO UNTIL (past end of table)
:   IF (current slot is an empty slot) THEN
:   :   (store pointer to symbol in symbol table)
:   :   (store current address)
:   :   (store type of reference)
:   :   (return)
:   ELSE
:   :   (increment to next slot in table)
:   ENDIF
ENDO
(signal error since forward reference table is full)
(return)
```

Listing 1: Set Forward Reference Routine. This routine is called once for each reference to a symbol that is not resolved in the symbol table and is not being used as a label.

Resolve Forward References

```
(start at beginning of forward reference table)
DO UNTIL (end of table)
:   IF (this reference to symbol table = current symb) THEN
:   :   (save current address)
:   :   (current address = address from forward reference table)
:   :   IF (relative addressing) THEN
:   :   :   (calculate relative offset)
:   :   :   (write it out)
:   :   ELSE
:   :   :   IF (two bytes required) THEN
:   :   :   :   (write out symbol's high byte)
:   :   :   ELSE
:   :   :   ENDIF
:   :   :   (write out symbol's low byte)
:   :   ENDIF
:   :   (clear table position for future use)
:   :   (restore current address)
:   ELSE
:   ENDIF
:   (increment to next table position)
ENDDO
(return)
```

Listing 2: Resolve Forward Reference Routine. This routine is executed each time a symbol flagged as unresolved in the symbol table is used as a label.

table is cleared for future use. Therefore, a source program may have as many forward references as it needs, but the number of forward references pending at any one time is limited by the size of the table. If any pointers to the symbol table are found at the end of the assembly, the undefined symbols are printed as potential errors. The Resolve Forward Reference routine has the logical structure shown in listing 2.

The third major change is a functional modification. A routine has already been defined to maintain the symbol table, but now it will also be required to recognize symbols that are unresolved, and to determine when the new functions developed above must be called. The first time an unresolved symbol is used, it will not be found in the symbol table, so it must be entered into the next available slot. If there is no available slot, a symbol table full error condition is raised. Set Forward Reference will then be called and a zero address returned. Additional references to this symbol will now be found in the table. Because each is flagged as unresolved in the symbol table, Set Forward Reference is again called and a zero address is returned. If the symbol is used as a label, the current address is entered into the symbol table and returned as the label's value. Resolve Forward Reference cannot be called at this time because the current instruction may modify the value of the symbol just resolved by equating it to a user defined value. Resolve Forward Reference must be called after the current line of source code has been completely processed. Now if another label for this symbol is encountered, a duplicate label error condition is raised.

To avoid using a separate byte of memory for the unresolved flag for each symbol in the symbol table, a zero address is used to indicate an unresolved symbol. This allows a zero value to be normally returned for such symbols as required by the above logic. The only effect this will have on source code programming is that a label at location zero cannot be resolved. Therefore, a symbol should not be defined at or equated to zero. Each reference to it would add an entry to the forward reference table and soon fill it up. Instead, a decimal value of zero should be used. The Maintain Symbol Table routine now has the logical structure shown in listing 3.

These functional modifications enable the assembler to completely process a source program in one pass. At this point, the functional requirements for processing with only one IO interface can be considered.

Logically, each program to be assembled has three basic sets of data to be handled: the source code used as input to the assembler, the object code developed by the assembler, and a program listing showing the object code that is generated for each line of source code. Physically, an IO interface is usually used to read and write data to or from one external device. The problem of handling three logical sets of data on one physical device has been solved by combining the generated object code with the listing. This reduces the data handling requirements to one input file (source code), and one output file (the combined listing) which the single interface can easily handle. The generated object code and its associated address that the loader uses are already produced on the left side of the listing as shown in listing 4. Therefore, it is only necessary to provide a way for the loader to directly read the listing to find the data that it needs.

How the listing is made available to the loader will vary with different physical configurations, and there are unlimited variations in possible hardware arrangements for small systems. This assembler was initially written to run on an ASR model Teletype

Maintain Symbol Table

```
(start at beginning of symbol table)
DO UNTIL (past end of table)
:   IF (current symbol = incoming symbol) THEN
:   :   IF (incoming symbol is a label) THEN
:   :   :   IF (table address = zero) THEN
:   :   :   :   (load symbol into table)
:   :   :   :   (set symbols to be resolved condition)
:   :   :   :   (return table address)
:   :   :   ELSE
:   :   :   :   (signal error, duplicate symbol definition)
:   :   :   :   (return)
:   :   :   ENDIF
:   :   ELSE
:   :   :   IF (table address = zero) THEN
:   :   :   :   (call SET FORWARD ADDRESS)
:   :   :   :   (return zero value)
:   :   :   ELSE
:   :   :   :   (return table address)
:   :   :   ENDIF
:   :   ENDIF
:   ELSE
:   :   IF (current table slot is an empty space) THEN
:   :   :   (load symbol into table)
:   :   :   IF (incoming symbol is a label) THEN
:   :   :   :   (return current address)
:   :   :   ELSE
:   :   :   :   (call SET FORWARD REFERENCE)
:   :   :   :   (return zero value)
:   :   :   ENDIF
:   :   ELSE
:   :   :   (increment to next slot in symbol table)
:   :   ENDIF
:   ENDIF
ENDDO
(signal error, symbol table full)
(return)
```

Listing 3: Process Symbol Table Routine. This routine is called each time a symbol is used. If the symbol has a resolved value it is returned to the calling module, otherwise a zero value is returned.

Sample Listing

```
LOCN  B1  B2  B3

0001  CE  00  20    >CLEAR  LDX  #TBL          LOAD IX WITH START OF TBL
0004  6F  00        >LOOP   CLR  X             CLEAR TBL LOCATION
0006  08            >       INX                INCR TO NXT TABLE POSN
0007  8C  00  30    >       CPX  #TBL+LENGTH   END OF TABLE?
000A  26  F8        >       BNE  LOOP          IF NOT, DO IT AGAIN
```

Listing 4a: Combined listing as seen by the user. The generated code shown under the headings B1, B2, and B3 are to be loaded starting at the address shown under LOCN. This ***was generated at BYTE using Jack's assembler as modified for Carl Helmers' homebrew system.***

with both tape reader and tape punch control characters enabled. A permanent machine readable copy of the listing can be produced on paper tape during the one pass operation. The listing, however, must be modified to enable the loader to find the required data when reading this tape.

In the 4 K version of this assembler, items that the loader needs can be surrounded by special nonprinting characters that can be recognized while scanning the listing. This isolates the needed items from source code, headings, comments and other extraneous items. These characters must not appear in any other context within the listing, so control characters (which are not valid assembler input) are the most logical choice. To the loader, then, the listing shown in listing 4a would appear as shown in listing 4b where the '(' and ')' represent nonprinting characters used to signify the start of an address, and the end of any bytes to load respectively. The high level form of the logical structure for a loader which would be able to use such a listing is shown in listing 5.

The combined assembly listing is produced by starting each line of source code in the middle of the page. The carriage return at the end of each line returns the current print position to the front of the line. The generated code is then printed in front of the source code. Before prompting for the next line of source code, a line feed is issued to position the paper for the next line. The result is most interesting to watch on the Teletype since one normally expects a line feed at the beginning or end of a line. The loader recognition characters are combined with "punch on" and "punch off" characters to keep all unwanted items off the object tape.

There are IO timing considerations worth mentioning that apply regardless of a system's configuration. If the input is coming from a tape file, input operations must stop to allow the generated code to be put out because the Motorola MIKBUG IO routines cannot support concurrent input and output. If stopping the input is not feasible for a given system, additional routines may have to be written to overcome the MIKBUG limitations. For interactive operation, there is rarely a problem as the assembler is usually faster than the user. For a truly minimum system, the source code can be entered by hand from the terminal, and the generated object code can be copied from the combined listing and loaded, again by hand. In this case, no intermediate files are required at all!

The functional changes necessary to allow the assembler to operate with only one IO interface are handled as modifications to the existing IO routine to enable it to create the listing as shown above. Specific changes may differ for various system configurations, and will not be shown here. We have now considered functional modifications to accommodate our requirements that the assembler operate in one pass, and that it use only one IO interface. The final requirement which will modify some of the original functions is that it run in only 4 K of memory.

The first thing to do is to look at all the functions available in existing M6800 assemblers and eliminate alternate ways of accomplishing things. In a tiny assembler, alternates are usually expendable overhead. For example, operands for the A and B accumulators may optionally be separated from the mnemonic instructions in 8 K assemblers (for example, LDA A instead of LDAA). This has been changed to require that the 'A' and 'B' be attached to the mnemonic which eliminates an additional scan for the accumulator operand. The choice of entering

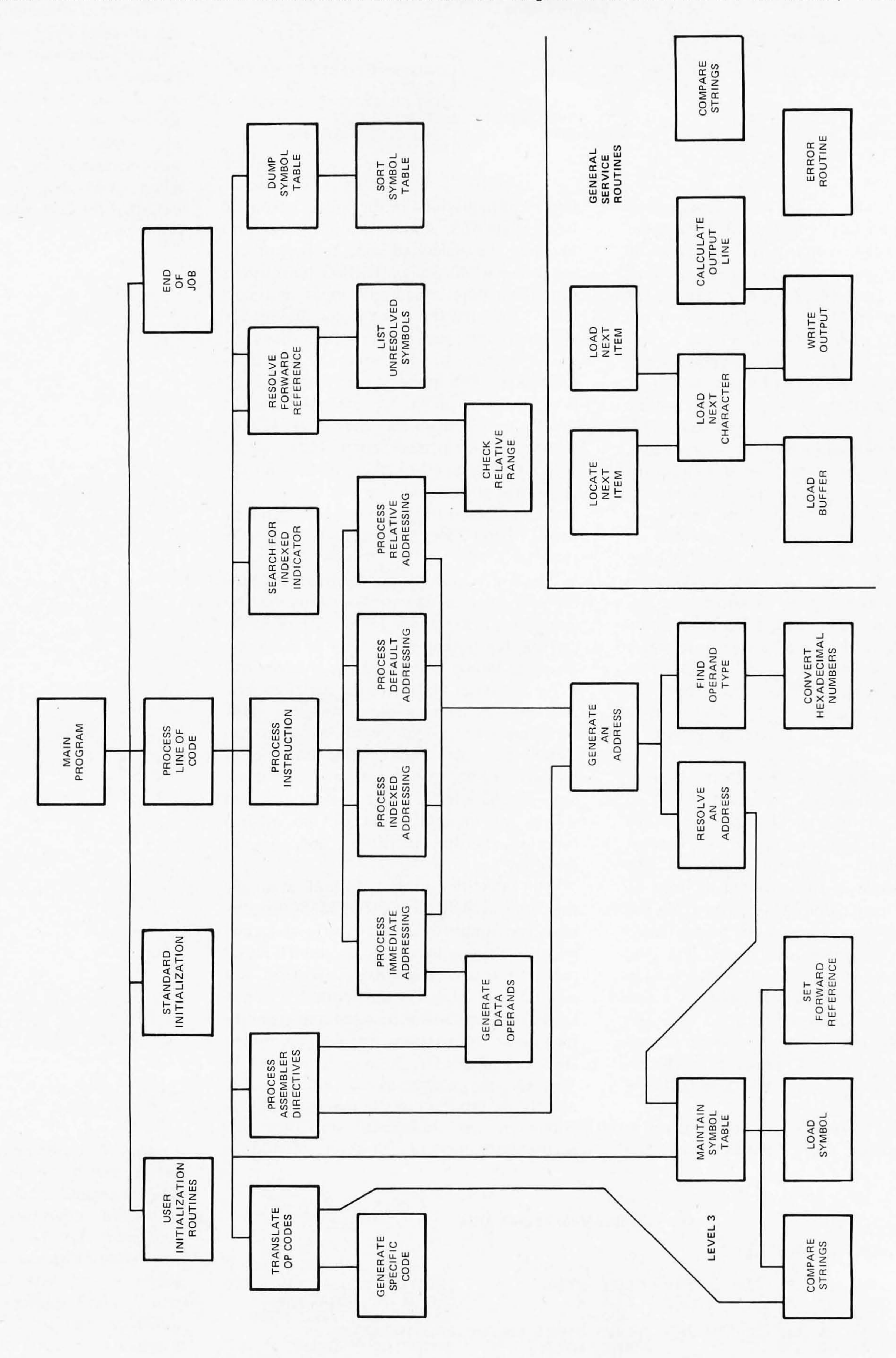
Loader's View of the Listing

```
LOCN  B1  B2  B3

(0001 CE  00  20)   >CLEAR  LDX  #TBL          LOAD IX WITH START OF TBL
(0004 6F  00)       >LOOP   CLR  X             CLEAR TBL LOCATION
(0006 08)           >       INX                INCR TO NXT TABLE POSN
(0007 8C  00  30)   >       CPX  #TBL+LENGTH   END OF TABLE?
(000A 26  F8)       >       BNE  LOOP          IF NOT, DO IT AGAIN
```

Listing 4b: Combined listing as seen by the loader. The generated code and associated addresses are separated from the rest of the listing by special characters shown here as '(' and ')'. This is a simulation based on part (a) of this listing.

Figure 3: The Final Assembler Structure. This is a detailed structure diagram of the assembler as it was actually written.



data in decimal, hexadecimal, octal or binary form has been reduced to decimal and hexadecimal. Alternate indicators of data type (eg: using a leading '$' or a trailing 'H' to indicate hexadecimal) are restricted to leading indicators to simplify parsing routines. Other such redundancies can be eliminated whenever a significant amount of code is needed to support multiple ways of doing the same thing.

The next area where space can be conserved is in table structure optimization. The largest table is the symbol table, so symbol size has been reduced from six characters to four. The first three and the last one character of a label of any size over four are combined as the character string entered into the table. This matches the four character size of the mnemonic plus accumulator instruction mentioned above, and the four character format of a hexadecimal address. The input routines may therefore be built around a standard four character format. Instead of using a binary search in the operand translation table, a hash index into the opcode table based on the mnemonic's first letter is used. *[See Terry Dolhoff's article on hashing techniques on page 18 of January 1977* BYTE.*]* This keeps the number of table entries searched to an average of 4.5 per opcode, and reduces the length of the stored mnemonic. By combining this access method with routines making use of the M6800's opcode structure, all of the 114 four character mnemonic instruction and assembler directives can be kept in a table keyed by 79 two character strings.

The next area where space can be conserved is in the simplification of complex routines at the possible cost of some efficiency or function. Because the symbol table in this assembler is smaller than those in full scale assemblers, and because we've got a dedicated personal machine to work with, complex hashing routines can be dropped in favor of a simple sequential search which burns a bit of computation time. Many complex error detection routines have been simplified to catch the most common occurrences of a problem. All physical IO operations are accomplished through the Motorola MIKBUG monitor rather than through uniquely written routines.

Loader Structure

```
DO UNTIL (end of listing)
:   DO UNTIL (character read is a '(')
:   :   (read character)
:   :   IF (character is null) THEN
:   :   :   (stop)
:   :   ELSE
:   :   ENDIF
:   ENDDO
:   (read address into index)
:   DO UNTIL (character read is a ')')
:   :   (read character)
:   :   IF (character is a blank) THEN
:   :   :   (read a byte from two hex characters)
:   :   :   (add incoming byte to indexed address)
:   :   :   IF (carry bit is set) THEN
:   :   :   :   (increment byte beyond indexed address)
:   :   :   ELSE
:   :   :   ENDIF
:   :   :   (increment index)
:   :   ELSE
:   :   ENDIF
:   ENDDO
ENDDO
(stop)
```

Listing 5: Logical structure for a directed loader capable of processing the listing shown in listing 4b. See listing next month for the implementation of a loader based in part on this structure.

The final method used to save space is to simply eliminate functions that are of limited use and require significant code to implement. There is no provision for multiplication or division in the expressions within an instruction's operand. Multiple additions or subtractions are possible, however, and can often be used where multiplication and division are usually found. There are no provisions for naming programs or for selecting options in generating the combined listing. Some error checking, such as invalid address or byte overflow, has been completely eliminated. The code generated is as compact as possible to fit in small machines, but is not relocatable. If it is desired in another location, it must be reassembled.

The result of all this compaction effort is an assembler whose basic executable code fits within 2170 bytes, and will run completely within 4096 bytes of programmable memory with a symbol table capacity of 200 symbols. Modifications have been made to almost all of the existing functions to accomplish the above changes. This completes the functional additions and modifications required for the assembler defined by the project specifications as shown in figure 3.■

In the next issue of BYTE, we'll continue Jack's article on his "Tiny Assembler" with his discussion of coding the assembler. The second part includes object code and information on patching and relocating the program.



THE FIRST WEST COAST COMPUTER FAIRE

A Conference & Exposition on Personal & Home Computers

FIND OUT ALL THE DETAILS –

GET YOUR **FREE** *COPY* OF THE **SILICON GULCH GAZETTE**

- Details of the programs & speakers
- Information about the banquets' arrangements
- Accommodations information, & registration forms for the St Francis Hotel (Faire Headquarters Hotel), & the other three Computer Faire hotels
- Pre-registration details for the Faire (reduced admission for those who pre-register)
- Details of the *Proceedings* expected to be published
- Articles about the homebrew exhibits
- Weather information, & tourist trivia regarding San Francisco
- Listings of the commercial exhibitors
- *AND*, just to make it interesting:
  - "Hot news", & raging rumors from "Silicon Valley" (the San Francisco Bay Area)
  - Product announcements, equipment descriptions, hardware & software news, etc.

CO-SPONSORS INCLUDE AMATEUR, PROFESSIONAL, & EDUCATIONAL GROUPS

- The two largest amateur computer groups:
  - Homebrew Computer Club
  - Southern California Computer Society (SCCS)
- Both Bay Area Chapters of the Association for Computing Machinery (ACM)
  - San Francisco Peninsula Chapter
  - Golden Gate Chapter
- IEEE Computer Society's Santa Clara Valley Chapter
- California Mathematics Council
- Stanford University's Electrical Engineering Dept.
- University of California's Lawrence Hall of Science
- People's Computer Company (PCC)
- Community Computer Center
- Bay Area Microcomputer Users Group (BAMUG)
- Professional & Technical Consultants Assn (PATCA)
- Amateur Research Center

april 15-17, 1977 · san francisco

Cut out, or photocopy; then fold, and tape in position.

☐ Send a *free* copy of *Silicon Gulch Gazette*.

☐ Send program & accommodations information, & Pre-registration Form (those who pre-register may take advantage of a reduced admission fee).

☐ Send speakers' instructions.

☐ Send homebrew exhibitors' instructions.

☐ Send commercial exhibitors' prospectus.

☐ Send details on the expected Computer Faire *Proceedings*.

PLEASE PRINT OR TYPE

NAME ______

ADDRESS ______

______

CITY ______ STATE ______ ZIP ______

77-1-12b

by COMPUTER FAIRE † (415) 851-7664

BUSINESS REPLY MAIL
No Postage Necessary If Mailed In U.S.A.

FIRST CLASS
Permit No. 91
Palo Alto, CA

POSTAGE WILL BE PAID BY

THE FIRST WEST COAST COMPUTER FAIRE

BOX 1579
PALO ALTO CA 94302

Establishing the CHU Dynasty

Let It All Hang Out

Stephen B Gray
Amateur Computer Society
260 Noroton Av
Darien CT 06820

Much time is taken up at computer club meetings, and especially at computer conventions, with people asking each other what machines they have, how much memory, what peripherals, and so on. Much time could be saved if there were some faster way of communicating this information, and one possibility is the Computer Hobbyist Uniform, or CHU, for the purposes of adding one more incomprehensible acronym to the vernacular.

The basic requirements for the CHU are that it be inexpensive, readily obtainable and lightweight. The one garment that fits all these criteria is the white lab jacket, with long sleeves and a button or two in front, also known as an intern's jacket, available everywhere for $9 to $22, with an average price of $13.

As for insignia, the most widely available type is military, and Army stripes are well suited to the job. The resulting combination provides a pleasant uniformity that helps make for a more homogenous group, as well as toning down the flashy dressers and improving the appearance of the slobs.

Stripes for Memory

Since many computerniks seem proudest of how much memory their machine has, it follows that Army noncom stripes are best used to indicate the amount of semiconductor storage. As figure 1 shows, the number of stripes indicates how much memory the wearer's computer has. For instance, if his Altair 8800b has at least 2 K bytes and as much as 4 K–1 (ie: 4095) bytes, he wears one stripe on each sleeve; from 4 K bytes to 8 K–1 bytes, two stripes; from 8192 bytes to 16,383 bytes, three stripes; and so on. The mathematically minded reader will note that the lower limit (in units of K=1024 bytes) is 2 raised to a power equal to the total number of stripes and rockers. So three stripes and two rockers indicate that the wearer has at least $2^5 = 32$ K bytes of semiconductor memory in his or her computer, and perhaps more.

For the memory fanatic with 128 K up to 256 K bytes, there's the first sergeant's insignia, with three stripes, three rockers, and a diamond. Incidentally, no stripes on the upper sleeve indicates up to 2 K–1 bytes of memory.

Service Stripes

A diagonal stripe on the lower left Army sleeve indicates three years of service. On the CHU, it indicates how many years the wearer has had his or her computer up and running. Although someone with an Altair 8800 might want each stripe to indicate a year, it's better to let the three year period stand. After all, anybody who's had a Scelbi 8H, or Mark-8, or RGS 008A running since they were first available will be entitled to wear a single three year stripe by Spring of 1977. And if the stripe were to stand for only a year, the CHU sleeve could get terribly cluttered by 1984.

There are other sleeve decorations that could be used, but which might only add to the clutter. There's the short horizontal bar for six months overseas, which might be used to indicate that the wearer has operated a hobby computer in a foreign country. The Unit Citation might be used by computer clubs to indicate some sort of group meritorious action, such as designing a computerized course selection system for the

local high school, or working out a method for beating the local roulette wheel operation to help supplement club funds, or calculating how much beer should be ordered for an all-night programming session.

Maker Patches

Now that your staff sergeant's stripes have identified you from a distance as a BMOC (Big Man on Computers), a shoulder patch, worn at the top of the left sleeve, identifies the maker of your computer.

These colorful patches, similar to the Army patches that identify a soldier as belonging to the Seventh Division or the Fifth Army, show the computer manufacturer's name in a distinctive and artistic design. Such patches could be sold by IMSAI, MITS, Southwest, etc, or packed with each computer kit. To make, they cost less than $1 each in large quantities.

A slight problem arises if a manufacturer offers more than one computer, such as the MITS Altair 8800b and 680b. There are two solutions for this ambiguity. A different patch could be made up for each, with the model number included; or a rectangular patch, bearing only the model number, could be made for sewing on right under the maker's patch.

Another problem comes up if the CHU wearer has, say, both an ETC-1000 and an EBKA 6502 Familiarizor. He or she can easily wear the ETC patch on one shoulder and the EBKA patch on the other. The question arises: Should the memory stripes show total memory, or should each set of stripes show only how much memory is in the computer whose maker's patch is on the same sleeve? It all depends on whether he or she minds having an unequal number of stripes on the two sleeves. Physicists, mathematicians and other lovers of symmetry may become emotionally hung up on this issue.

If the CHU wearer has more than two computers, he or she has two solutions: either stacking the makers' patches one below the other on the sleeve, or sewing the extra ones on the front of the jacket — but not just above the breast pocket area, because that's where we put the Peripheral Pins and Data Decorations.

Peripheral Pins

Now that you've been identified from afar as the owner of a Southwest 6800 with 8 K bytes of memory, something is needed to show what you've got attached to your computer. A simple solution is to use zapped 14 pin or 16 pin ICs, lettered in white, inserted into a narrow strip of P pattern

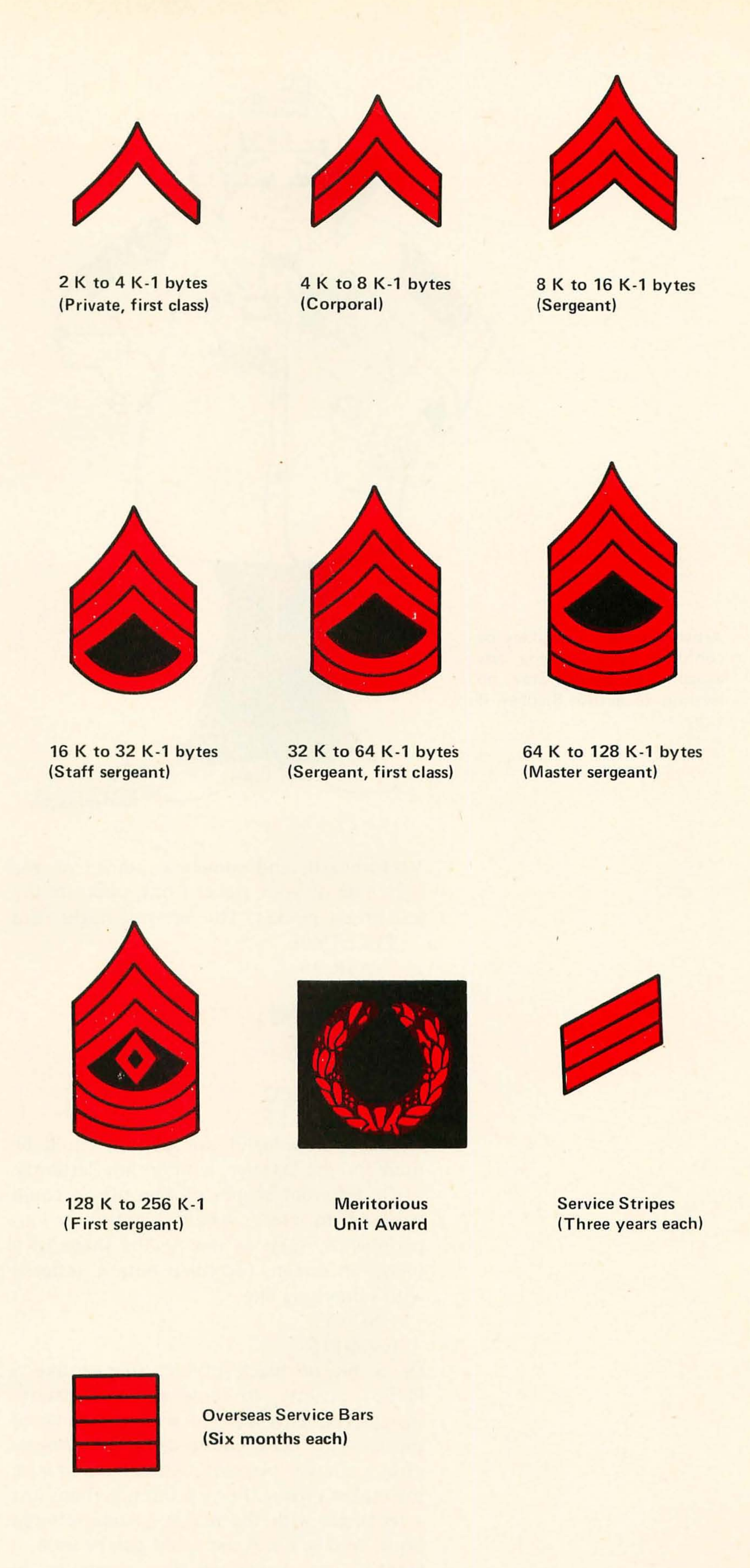


Figure 1: Representative military stripes adapted with semantic redefinition to help CHU computer addicts.



Artwork credit: Lest there be confusion, this drawing was executed by Sandra Gray, no relation to author Stephen B Gray.

Vectorboard, and somehow pinned on the right side of your jacket front, opposite the left breast pocket. The lettering might read

TELETYPE
ASR 33

or

SOUTHWEST
CT-1024

or even

CROMEMCO
TV DAZZLER

Actually, a colorful dot pattern could be used for the Dazzler, without any lettering.

Or different shapes of thin plastic could be used to more readily identify various peripherals, such as one in the shape of a television screen, for video output, lettered with something like

SONY
CVM-115

Or a bit of black plastic, shaped like a Philips cassette, to show you use cassette storage, perhaps lettered with the name of the cassette deck you're using. For floppy disk, a circular piece of plastic, lettered with the maker's name. For a printer, perhaps just a rectangle with the maker's name lettered on it. And so on. A pin could also be used to identify the maker of your computer, in addition to the shoulder patch, for faster face-to-face identification at club meetings.

Name Tags

Name tags are often used in the Army. The CHU version is a dead MPU IC, without pins, with your name lettered on it, worn above the Peripheral Pins over where the right breast pocket would be if lab jackets had one. (Dan Meyer, head of Southwest Technical, wore a 6800 MPU IC on his shirt at Personal Computing 76 in Atlantic City last fall, with DAN and SWTPC lettered on it, partly inspiring this article.)

Data Decorations

The area above the left breast pocket is reserved for Data Decorations, which are for meritorious action above and beyond the call of duty. There's the black-and-blue ribbon for those who successfully fought the "Battle of the 66 Wires" on the original Altair 8800, and the very rare blue, grey and brown ribbon worn by that gallant handful of pioneers who captured Hill 680, only to lose it when it was retaken.

There are also software decorations, although this area is somewhat amorphous, since what is a difficult feat for one-hobbyist might be a snap for another, just as in hardware achievements. It's up to each hobbyist to decide if he or she is enough of a hero to merit this or that Data Decoration. There might be a software ribbon for the person who wrote his or her own assembler, or Tiny BASIC, etc.

If a ribbon refers to a numbered piece of equipment, the resistor color code can be used for the ribbon colors, such as blue, grey and brown for the 680. Cryptic types can simply wear a 680 ohm resistor, attached to Vectorboard.

Scratch Built and Homebrew

There are some situations not covered so far. What about the hobbyist who prefers to build his or her computer from scratch? What does he or she use for a maker's patch? If there's enough demand for them, they could be supplied by the integrated circuit manufacturers, such as Intel, Motorola, Zilog, etc. Or again, by a computer store chain.

For the real hard-core scratch builder, who scorns microprocessors as the easy way out and designs his or her own processor from TTL or ECL integrated circuits, a "homebrew" patch could be designed, perhaps on the style of a mountaineer's still, or witch's cauldron.

Far Out

For the hobbyist who is *really* into computers, and especially vice versa, the

WE HAVE IT. THEY DON'T.

SOFTWARE

DOS

Disk Operating System *Device management and executive scheduling *Directoried file management system *Sequential and indexed random file structure *Full support of all languages, utilities, and peripherals.

FORTRAN IV

ANSI FORTRAN compiler *Double precision real and complex arithmetic *Non zero based arrays *reentrant subroutines *Random file I/O *ISA process control, real time, and multitasking extensions *Chaining, swapping, and overlays *Commercial subroutine package.

MACRO ASSEMBLER

Relocatable code generation *Macros *Dynamic instruction definitions *Conditional assembly.

EDITOR

Character or line oriented *String operations for search, search and replace *Store and execute command strings *Multiple files open concurrently.

HARDWARE

CPU

Micro NOVA® 16 bit processor *Real time clock *Hardware multiply and divide *Power fail auto restart *8K bytes RAM *9 slot chassis with front panel and power supply. (optional 18 slot chassis)

ASYNCHRONOUS SERIAL INTERFACE

Strap selectable data, parity, and stop bits *Modem control signals *Data rate strappable from 50 to 19,200 baud *EIA-RS232C or 20ma interface.

DEBUG ROM

Examine, deposit, control of memory and internal CPU registers and status *Breakpoints for monitoring running programs *Auto program load.

DISK

2 diskette drives per controller *315K bytes per diskette *31.5K bytes per second transfer rate.

EXPANSION

Memory expansion in 8K and 16K byte increments up to 64K bytes *General purpose interfaces for custom applications *9 slot I/O expansion chassis.

DISK SYSTEM

*CPU with 40K bytes RAM *Dual diskette *Asynchronous serial interface *CRT terminal *Debug ROM *DOS *FORTRAN IV *Macro Assembler *Editor *Utilities and support programs *Complete documentation set. **$8400**

Other systems available from **$2700**
All hardware is fully assembled and tested. Support is available from service centers around the country. Prices and specifications subject to change without notice.



BPI

If you are serious about computing, call or write today. BPI Inc., 2205 East Broadway, Suite 6, Tucson, Arizona 85719 (602) 326-6975

® NOVA is a registered trademark of Data General Corporation, Southboro, Massachusetts.

ultimate CHU add-on is a 100 pin printed circuit board edge, protruding from the back of the jacket, just above the shoulder blades.

For the lunatic fringe, there's the computerized CHU, or COMCHU, with a SC/MP or other small board in one of the side pockets, controlling the flashing of LEDs and other devices inside insignia or pins or whatever, producing a Christmas tree effect. CAUTION: The COMCHU, if worn in public, is a sure ticket to a rubber room.

Other Insignia

Many other types of insignia could be borrowed from the military, but again, they would only clutter up the CHU. There's the fourragère, a braided cord worn around the left shoulder seam as a unit decoration. There's the Combat Infantry Badge, worn above the ribbons over the left breast pocket. And there're branch of service pins, worn on the lapel to indicate Infantry, Corps of Engineers, Ordnance, etc. With a little imagination, these could all be used, but the effect would be more gaudy than anything else. This applies especially to medals, which are expensive to make; it's much neater to wear the equivalent ribbons.

Same for officer's insignia, such as bars, oak leaves, eagles and stars. They look good, but not when mixed with enlisted persons' stripes. For those who insist, such insignia might be used to indicate rank or seniority in a computer club.

Other Jackets

The hawks among us may prefer a military jacket, to go along with the military insignia, rather than a lab jacket. The catch is that a GI jacket is apt to cost more, and isn't as easily and widely available. And it looks rather odd if worn with civilian trousers. If you want to go whole hog and wear uniform trousers and jacket, along with stripes and decorations, you might get in trouble with the MPs. However, if a full uniform is your thing, then you might go all the way. Get a web belt, a holster, and start toting a Replica Models copy of a .45 automatic, with a light pen cleverly built into the barrel. The ammo pouches could be used to hold program cassettes for trading with other software thieves.

Headgear

Since interns and lab technicians seldom if ever wear hats, almost any type of headgear would look odd with a CHU. However, continuing in the military vein, there is one item that might appeal to the same lunatic fringe type who has a COMCHU, and especially to those who prefer a GI jacket and trousers.

A copy of an Army plastic helmet liner could be made of the same material used in PC boards, printed wiring could be applied, and the whole thing turned into something like the battery powered Intersil Intercept Jr, with keyboard on top, display in front, and batteries in the rear. The curved surfaces might be too much of a challenge to a PC board designer, so perhaps some of the curves might have to flattened. This would make the CHO (Computer Helmet, Official) look even more futuristic and science fictionish. The full uniform CHU/CHO combination would be enough to stop a digital clock, as well as get you in trouble if worn outdoors.

Belt Buckles and Tie Pins

The CHU is meant only for wearing to computer club meetings or to computer conventions, as it's a little too odd to wear in public. However, there are several ways of letting people know you're a computernik, without the CHU. And they can be worn with the CHU, too.

If there's enough demand for them, belt buckle equivalents of the maker's shoulder patches could be offered by MITS, Polymorphic, OSI, etc. These can be either the flat laminated plastic multicolored type, or cast metal bas relief. Or the belt buckle could be a miniature replica of the front panel of your Astral 2000 or PCM-12, or a replica of a SC/MP or KIM-1 in its entirety, either as a laminated photoreduction, or in cast metal. The Cromemco TV Dazzler would be a natural for a multicolored mini-photo laminated belt buckle.

Tie pins offer a variety of possibilities. Perhaps a miniature Continental Specialties LP-1 Logic Probe, or an 8080 processor, or a DB-25 connector. Or, once again, a miniature of a front panel or a PC board.

The CHU Counter

Although maker's patches are perhaps best supplied by the computer manufacturers, all the rest of the CHU insignia could easily be made up by one of the computer store chains, or by one of the manufacturers of military insignia. Computer stores could have a CHU counter, or CHU shelf, displaying the various stripes, decorations and pins. The best display would be a CHU in maximum configuration, with jacket, full set of stripes, chestful of pins, and a bouquet of decorations.■

Deal Yourself in...



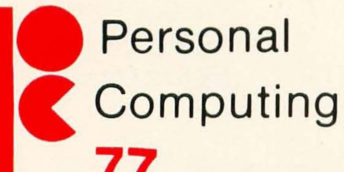
Atlantic City, N.J.
August 27th-28th

What it's all about!

Software Development
Micro Computers
Hardware Development
Disc Memories
Computer Comparisons
Interfacing
Program Implementation
AMSAT
Computerized Music
Video Terminals
Kit Construction
Printers
Computer Games
Digital Tapes

Special group rates for clubs and organizations traveling from the West Coast and Mid-America contact:
7 Seas Travel
17220 S. Norwalk Blvd.
Cerritos CA 90701
Dawn Corrigan
213-924-8383

- Seminars and Technical talks by leading electronic equipment manufacturers
- Major Exhibits from all over the country
- Demonstrations in many areas including Home and Personal Computing
- Door Prizes, Free Literature and Free Mementos
- All this plus Sun and Surf - Fun and Excitement - Relaxation and Leisure



Consumer Trade Fair

Write for FREE TRIP-KIT to PERSONAL COMPUTING 77, Rt 1, Box 242, Mays Landing, New Jersey 08330

EXHIBITION BOOTHS STILL AVAILABLE - CALL (609) 653-1188

Reviews

A New Journal for Computer Music Lovers . . .

John Snell, a computer person with a specialization in music, has just decided to start a new journal of information for the computer music person. This journal is called, quite naturally, the *Computer Music Journal* and is published by John through the facilities of PCC. The address for the journal is 1010 Doyle St, Box E, Menlo Park CA 94025. According to John's letter, which accompanied some descriptive copy, the first issue was expected early in 1977. The following is John's description, quoted from his letter:

> *The first issue of* Computer Music Journal *includes an excellent comprehensive introductory article on computer music in general, written by Andy Moorer at the Stanford AI laboratory computer music project. Also included is an article by Dexter Morrill on high quality trumpet algorithms. I will print several of my own articles in this issue. One is about the design of a digital oscillator controlled by a small computer. It can output up to 256 low distortion sine waves with independent control of amplitude, frequency, and phase. I will also print an article on high speed multiplication. I have designed a TTL (74LSXX) circuit which will compute a 16 to 32 bit product every 75 ns. I may also include a computer music reading list if there is room. I will also probably print an article by Steve Saunders on multiplicationless FM synthesis of natural timbre.*

The price of a subscription to the *Computer Music Journal* is $14 per annum. We've already signed up for one and expect considerable interest on the part of BYTE readers in general.■

Early Indications of Technology in Roman Military Arts or Plexitus

E E Barnes
Star Route
Parkesburg PA 19365

Sometimes an individual is about to bring about an invention which could change the world, but is never noticed. We often say that this man was born before his time. Omnius Plexitus, legate of the XIII legion, was such a man. Stationed with his legion in the rough country which later became Bulgaria, he improvised in order to survive.

Genius struck like lightning one day as he watched his artillery crews practicing. One of the pieces of semi-portable Roman siege equipment was called the onager or ass, because of its kick; this was a sort of huge crossbow which fired a heavy bolt or arrow perhaps six feet long. During this particular practice session, the cable release mechanism became stuck and a legionnaire climbed atop the device to free it. Free it he did, but only after he was astride the bolt to obtain leverage. Arrow and man sailed out as one across the treetops. History does not record the fate of this early pioneer, but this is the first historical incident of a man going aloft on an unpowered, heavier than air contrivance. This observation was not forgotten by Plexitus.

Plexitus was faced with supply problems in the hilly terrain in which his men were operating. They were widely spread out over the backs of ridges and across gorges. Supplying these detachments became rather serious because the number of men which had to be dedicated to carrying food seriously reduced those carrying spears. Plexitus solved this problem by attaching food to the lightened onager bolts and having these fired at the isolated detachments of his legion. This was moderately successful since his casualties were light and the process spurred other developments. Unfortunately the fact that he had fired upon his own men was used later during his court-martial in Rome.

This Idea Has a Lot of Impact

Plexitus reasoned that such a bolt should be able to carry a man, but the major difficulty he faced seemed to be predicting the point of impact of the missile (and man). After devoting much study to the problem, he was able to define 28 variables which directly influenced the flight of the missile. There was basically no limitation upon payload size; he merely built a bigger crossbow.[1] He had an onager scaled up and one was built concurrent with his attempts at aiming.

The first man known to fly was an aide to Plexitus, Dominic Idiotus. Idiotus (from whence comes the common technical term) was apparently offered a transfer if he would agree to attempt a flight in the Plexitus machine. In July of 248 BC he mounted the bolt and was propelled 280 yards into a local

[1] During the supply operation early in the development of the Plexitus machine, some of the troops received their pay in the same way as they did their food, attached to the onager bolts. A few scholars trace the roots of the word "payload" to this event.



swamp, from which he returned dirty but smiling. Exit Idiotus.

During the fall of 248, the machine was refined and experimentation continued. Plexitus found that most of the variables affecting his machine were products appearing in powers of one or more. (This in itself is a major accomplishment considering the state of instrumentation with which he had to contend. For example, humidity was found to change the tension of the bow string. The only means the Romans had for measuring humidity was by counting the number of crows made by a cock in the early morning. Plexitus apparently ran through a lot of chickens in his calibration attempts. One of his men complained in a letter of all the chicken the men were forced to consume.[2]) Anyone who has ever attempted to multiply in Roman numbers will appreciate the difficulty of raising a number to a power greater than one and then finding the product of 28 of them.

The Roman Abacus

He finally resolved this difficulty by moving over half his legion to a huge space which was cleared for this purpose and using men as markers in what became a kind of huge abacus. Men moved about in position upon orders from their squad leaders. Primitive, but it worked.

Unfortunately, the calculation took over eight days because of its complexity and some errors resulted from men in the morning not remembering the positions which they had held the night before. Plexitus proceeded undaunted and merely repeated the calculation and then took an average of the different answers which he obtained.[3] His work was interrupted several times by the hillsmen who took unfair advantage of his scientific preoccupation by attacking. It was during this first retreat that the first Plexitus computer was lost to the numerous but unappreciative locals.

Thus Plexitus was the first to use the abacus in the West. Unfortunately, it lacked portability, requiring over 7000 men and 34 acres in its operation. However, this is not the disadvantage that moderns might think since the ancients had plenty of time and space. A pre-Christian timeshare facility could have developed where Romans could have taken their problems to the machine,

[2] Some believe that later Roman soothsayers who attempted to predict the future by looking at the intestines of freshly slain chickens were attempting to reproduce the actions of Plexitus' computer, but were working from garbled, word-of-mouth reports.

[3] In standing all day, the legionnaires apparently took to shifting from one foot to the other, and when not actively moving about, to digging with the toes of their sandals in the sand. This was quite useful the next morning in determining the position the men had held the night before. This is the first mention of permanent memory in the literature.

The holes in the calculating field suffered from one operational disadvantage. When it rained, the holes filled with water and the men had a distaste for standing in water, so he would begin the day with the complement of the number which the field had held at closing time the night before. Plexitus outlawed the scuffling with the toes once he found the reason for his rather curious answers. Even good things had their problems. Some say that the water was not the reason for Plexitus outlawing this practice, rather it was traced to a comment made by a centurion who had a villa near Actium. This individual observed that this looked like an efficient way of plowing a field and would he do some calculations on his land near Actium next spring.

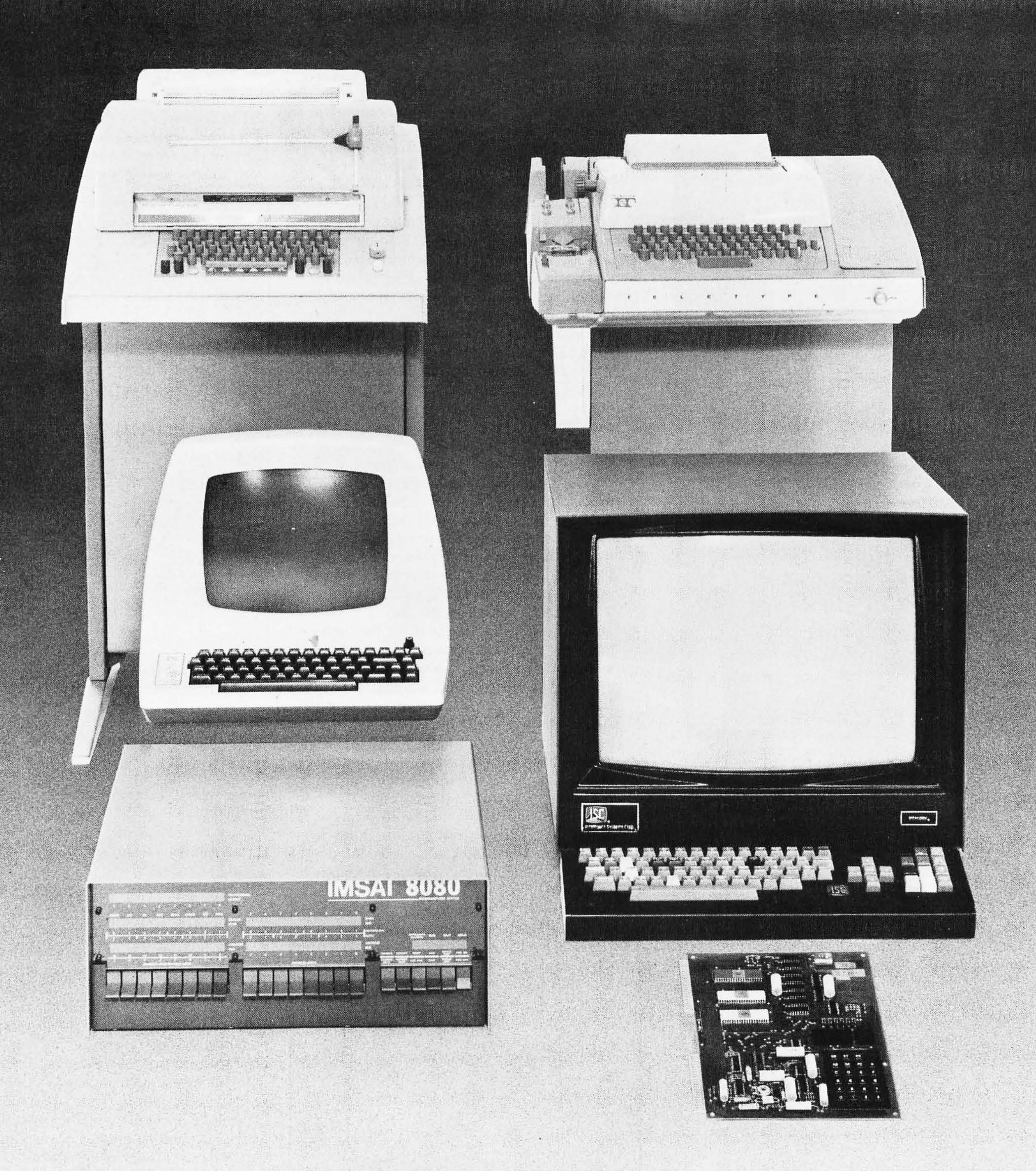
much like the Greeks did at Delphi. There are even some who suggest that Pompeii was attempting to use one of the Plexitus facilities during the later Roman civil wars; even this group is splintered with one faction saying that he never got it working. The other faction suggests that in fact he did, since he lost the battle and the war to Caesar.

During this time the XIII legion was forced to fight for its life, since the experiments had weakened the men; an individual does not use the same skills standing motionless in a field all day as he does carrying a shield and sword. About the time Plexitus had lost half of the province under his command, he was finally able to complete a second super crossbow. This machine he used to launch several groups of men in sharp, tight counter attacks behind the lines of the enemy. These were quit

the direct result of esprit de corps while others blame it on a reactionary attitude of those who did the calculations. Although normally the second log could be saved by quick action on the part of Plexitus, much time was lost. There are rumors among some scholars of this period who trace the battering ram to these acts, but this cannot be supported; we only know that Roman numerals were not normally carved in them, at least not one to a million.[4]

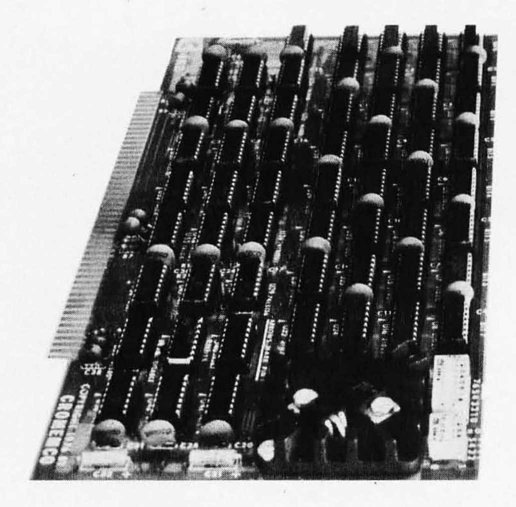
The Last Days of Plexitus

Plexitus, because of overwhelming odds, had now lost nearly all of the province which he had been sent to manage and was now with his back against the sea. He then launched his most extensive airborne operation against the Bulgarians, firing over half a thousand men across the enemy lines, fully


IMSAI 8080



Anticipation is Half the Fun



How do you make a relatively slow memory part run fast enough to keep a 4 MHz Z-80 processor busy most of the time? If you use a little ingenuity such as that found in Cromemco's new 4 K static programmable memory card you'll employ some coordination of the processor wait line with what computer designers call "address anticipation logic." The key to this address anticipation strategy is to assume that in most cases the next address generated to a given bank of memory will be the previous address plus 1. This is certainly true in programs when executing contiguous segments of code between jumps; it is likely to be true in many data access situations where sequential operations such as linear searches, block moves and block IO transfers reference a continguous block of memory space. So what the Cromemco card does is to assume that in any given 4 K bank of memory implemented with its new 4KZ card, the next address will always be the old address plus 1. On this assumption the card starts a memory cycle ahead of the availability of the address from the processor using a local 12 bit address counter. If it turns out that the processor address in the low order 12 bits is the same as the anticipated value, then there can be immediate access and completion of the memory cycle for the 21L02 circuits used in the board – with no processor wait states. If on the other hand the next address is not equal to the anticipation address, a wait state is generated and the counter is updated with the new nonsequential low order address value. If used with the fast Z-80 processor card which Cromemco also makes, there will thus be a mix of slow and fast cycles depending upon whether or not the next low order address reference to the card is one plus the old address. When used with a slow 8080 or Z-80 at 2 MHz, no wait states are ever generated since the memories are always fast enough.

The new Cromemco memory card is designed for convenient installation at any 4 K byte memory address boundary, with bank address selection via a slide switch mounted on the board (a bit more convenient than using soldered jumper wires.) Called the 4KZ board, this product is available in kit form from computer stores or the factory at $195, or assembled for $295. Delivery is quoted at five days after receipt of order. Cromemco is located at 2432 Charleston Rd, Mountain View CA 94043.■

THE SLOPPY DISK DRIVE

I thought you would like to know we have a computer store in Cincinnati with a rather interesting article: a sloppy disk drive.

I assume it is used with PPP or NNN transistors, "surprise pak" DIPs, and the famous Signetics WOM.

Perhaps some of your readers would have other precision (10^6 ppm) uses for it.

Sincerely on the lookout for such gems.

William R Stock
1125 Lois Dr
Cincinnati OH 45237

Documentation: The following clipping was found in the Cincinnati Enquirer, *Jan 13 1977, on page D-1:*

Make-it-yourself computer kits, books and printer and video kits are sold at Digital Design, 7696 Camargo Rd. Instructions are available on how to put the computer kits together. There is software, sloppy disc drives and almost anything the novice or professional would need. The shop is open from 9 a.m.-5 p.m. Mondays-Saturdays or by appointment.

How to Make the Z-80 Slave to a NOVA

SAI Comsystems Corp, a subsidiary of Science Applications Inc, 4060 Sorrento Valley Blvd, San Diego CA 92121, recently announced the availability of a fast cross-assembler for the Zilog Z-80 microcomputer chip that runs on the Data General NOVA and ECLIPSE line of computers under the RDOS operating system. This cross-assembler was written in assembly language for Scientific Applications Inc to allow speedy assembly of large, in-house, Z-80 systems programs used in new intelligent terminals developed by the firm.■

Get Yourself a Z-80 Reference Card

Fred McNeill, Box 6305, Denver CO 80206, sends us a copy of this Z-80 processor reference card which he has constructed from the Zilog documentation, and is selling for $3. The card is

Z-80* REFERENCE CARD

FLAGS - MNEMONICS - OPERATION CODES -

FLAG REGISTER→ INSTRUCTION ↓	S 7	Z 6	0 5	H 4	0 3	P/V 2	N 1	C 0
ADD s;ADC s	D	D	0	D	0	V	0	D
SUB s;SBC s;	D	D	0	D	0	V	1	D
CP s;NEG	D	D	0	D	0	V	1	D
AND s	D	D	0	1	0	P	0	0
OR s;XOR s	D	D	0	0	0	P	0	0
INC s	D	D	0	D	0	V	0	N
DEC s	D	D	0	D	0	V	1	N
ADD K,ss	N	N	0	X	0	N	0	D
ADC HL,ss	D	D	0	X	0	V	0	D
SBC HL,ss	D	D	0	X	0	V	1	D
RLA;RLCA	N	N	0	0	0	N	0	D
RRA;RRCA	N	N	0	0	0	N	0	D
RL;RLC;RR;RRC	D	D	0	0	0	P	0	D
SLA;SRA;SRL	D	D	0	0	0	P	0	D
RLD;RRD	D	D	0	0	0	P	0	N
DAA	D	D	0	D	0	P	N	D
CPL	N	N	0	1	0	N	1	N
SCF	N	N	0	0	0	N	0	1
CCF	N	N	0	X	0	N	0	D
IN r,(C)	D	D	0	0	0	P	0	N
INI;IND	D	D	0	X	0	X	1	N
OUTI;OUTD	D	D	0	X	0	X	1	N
INIR;INDR	X	1	0	X	0	X	1	N
OTIR;OTDR	X	1	0	X	0	X	1	N
LDI;LDD	X	X	0	0	0	D	0	N
LDIR;LDDR	X	X	0	0	0	0	0	N
CPI;CPIR	X	D	0	X	0	D	1	N
CPD;CPDR	X	D	0	X	0	D	1	N
LD A,I	D	D	0	0	0	IF	0	N
LD A,R	D	D	0	0	0	IF	0	N
BIT b,s	X	D	0	1	0	X	0	N

Subscript "L" - the low order 8-bits of a 16-bit register
Subscript "H" - the high order 8-bits of a 16-bit register
( ) - the contents within the ( ) are to be used as a pointer to a memory location or I/O port number
8-bit registers are A,B,C,D,E,H,L,I and R
16-bit register pairs are AF,BC,DE and HL
16-bit registers are SP,PC,IX and IY
IF - the content of the interrupt enable flip-flop is copied into the P/V flag.

FORMATS - SYMBOLIC OPERATIONS

→F REGISTER
→BIT LOCATION

FLAG DESCRIPTION	FLAG=1	FLAG=0
S=Sign of Results	- (M)	+ (P)
Z=Result is zero	=0 (Z)	≠ 0 (NZ)
H=Half Carry	Carry	No Carry
P=Parity(Log.)	Even(PE)	Odd(PO)
V=Overflow(Arith)	Over	Not over
N=Add/Subtract	Subtracted	Added
C=Carry from Results	Carry (C)	No carry (NC)

SYMBOL MEANING

D - Flag Depends on results
N - Flag is Not changed
0 - Flag is reset
1 - Flag is set
X - Flag is meaningless
V - Flag depends on oVerflow
P - Flag depends on Parity

DIRECTIONS:
To find Op Code from mnemonic, find mnemonic in alphabetic table. Octal Op Code is given immediately, or table in which it can be found. Mnemonics in tables 1, 2 or 3 are located at line and column numbers which juxtaposed give octal base Op Code and format. Format table gives 9 different formats, construct Op Code accordingly. Mnemonics involving K or M are expanded and clarified in small charts to left of tables.

To find mnemonic from Op Code the various prefixes must be worked through to find the correct table and base Op Code, the line and column number of the mnemonic then consists of the first two and third numbers of the base Op Code in octal, the symbolic operation of the mnemonic can then be found in the alphabetic table.

With tables toward you, hold left edge of card in left hand; tilting card toward you with left hand will bring flag and alphabetic lists into reading position.

*Trademark Zilog Corp.
Z 80 Card, Copyright 1976, Fred McNeill   Box 6305, Denver, Co.80206 $3.00

printed on heavy letter size stock with a fold. The number base orientation of the card is octal, and the information is carried in somewhat condensed form, so some mental gymnastics are required to use it.

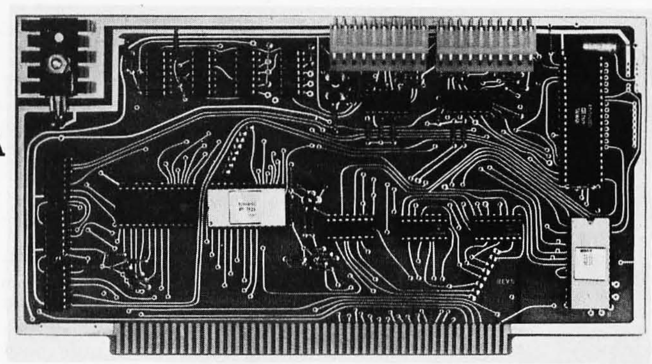
If you use a Z-80, you may find this card a handy accessory.■

Where to Get a 3M Drive . . .

The National Multiplex Corporation, 3474 Rand Av, Box 288, South Plainfield NJ 07080, is presently advertising an inexpensive 3M drive product for the personal computing market. The drive is approximately one fourth the price of new drives previously available, and operates with phase encoded data at rates of up to 9600 bps. Motor control, record and play mode are under control of software in your computer or a set of switches on the device. The device interfaces to any computer with a serial IO port. The current price of $220 includes electronics of the phase encoder board (accepts clocks at 16 times or one times the data rate), and motor control. According to the press release, delivery will be four weeks after receipt of order. At 9600 bps, with an 8 ips (20 cm per second) tape speed, the 300 feet (91.4 meters) of tape on this drive can be read or written (assuming no block overhead) in 450 seconds (7.5 minutes). Multiplying 450 seconds by 9600 bits per second and again assuming no formatting overhead, the raw bit capacity of this drive is 4.32 million bits per track, or a cartridge capacity of 17.3 million bits on four tracks. Dividing



DIGITAL DATA RECORDERS USING 3M DATA CARTRIDGES



MODEL 3M3 $199.95
(Price Increases to $220.00 effective 1 April 77)

2SIO(R) CONTROLLER $190.00

MODEL 3M3 Featuring the radically new "Uniboard" method of construction for data cartridge drives. The major computer makers are changing to cartridges at a rapid pace because of the freedom from binding and greater data reliability. Operates in the phase encoded self-clocking mode which provides greatly enhanced freedom from speed variation problems and allows 100% tape interchangeability between units.

Uses the 3M Data Cartridge, model DC 300. This cartridge contains 300 feet of .250 tape in a sealed plastic container. Using four tracks you can record nearly 2 megabytes of data on a cartridge.

Specifications: Full software control of record, play, fast forward and rewind. LED indicates interrecord gaps. EOT and BOT are sensed and automatically shut down recorder. Feedback signals send reset and inter-record gap signals back to the computer so that software searching for inter-record gaps at high speed can be accomplished. Can also be operated manually by means of the switches on top which parallel the software control signals.
$199.95 until April 1, 77. $220.00 after April 1, 77. Includes Phase Encoder Board (ACI)

For 8080, 8085 and Z-80 users

Comes complete with software listing for the programs in the 2SIO(R) ROMs. Can be controlled by any of the commonly used I/O boards. Send for complete documentation and interfacing instructions on 3M3 and 2SIO(R) ($3.00). These programs provide full software control.

2SIO(R) CONTROLLER (Bootstrap Eliminator)

This is a complete 8080, 8085, or Z80 system controller. It provides the terminal I/O (RS232, 20 ma., or TTL) and the data cartridge I/O, plus the motor controlling parallel I/O latches. One kilobyte of on board ROM provides turn on and go control of your Altair or IMSAI. No more bootstrapping. Loads and Dumps memory in Hex on the terminal, formats tape cartridge files, has word processing and paper tape routines. Best of all, it has the search routines to locate files and records by means of six, five and four letter strings. Just type in the file name and the recorder and software do the rest. Can be used in the BySync (IBM), BiPhase (Phase Encoded) or NRZ modes with suitable recorders and interfaces.

$190, wired and tested. – $160 0, Kit form.

Audio Cassette Interface (ACI) This is the phase encoding board used in the 3M3. Additional components on the board enable you to use audio recorders in the KC standard or the new PE 2400 (2400 baud) systems. Can also be used for Tarbell if you have an 8251 Intel I/O chip. Required if you use an audio cassette with the 2SIO(R) above.

$50, wired and tested. – $35, Kit form.

For 6800 Users: Software programs and I/O board for SWTP are under development. Limited software available now. Ask for 6800 data with $3.00 Documentation package. These programs will provide full software control.

CARTRIDGE AVAILABILITY: Cartridges are made by 3M, ITC, Wabash and others. They are available at all computer supply houses and most major computer service centers. We can supply them at normal current list prices.

"COMPUTER AID" and "UNIBOARD" are trademarks of the NATIONAL MULTIPLEX CORPORATION. The 3M Data Cartridges are covered by 3M Patents and Marks. "UNIBOARD" Patents Pending.

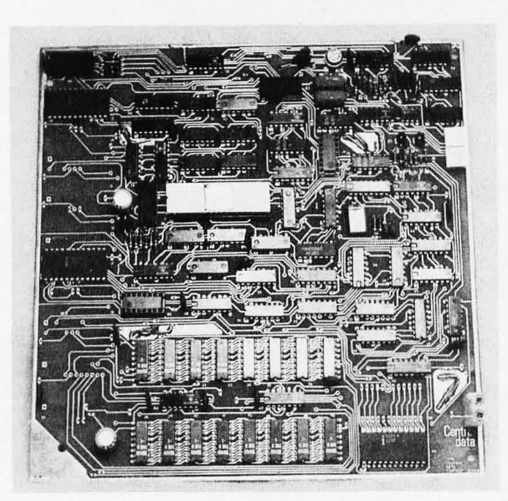
OVERSEAS: EXPORT VERSION- 220 V - 50 Hz. Write Factory or: megatron, 8011 Putzbrunn, Munchen, Germany; Nippon Automation 5-16-7 Shiba, Minato-Ku, Tokyo; Hobby Data, FACK 20012 Malmo, Sweden; G.Ashbee, 172 Ifield Road, London SW 10-9 AG; The Computer Place, 186 Queen St. W., Toronto, Ontario M5V 1Z1, Canada.

For U.P.S. delivery, add $3.00. Overseas and air shipments charges collect. N.J. Residents add 5% Sales Tax. WRITE or CALL for further information. Phone Orders on Master Charge and BankAmericard accepted.

NATIONAL MULTIPLEX CORPORATION
3474 Rand Avenue, South Plainfield, N.J. 07080, Box 288 Phone (201) 561-3600 TWX 710–997-9530

4.32 million bits per track by the 3600 inches per track, the density is 1200 bits per inch (470 bits per cm) on each track. Dividing 17.3 million bits by 8 bits per byte, again assuming no overhead, the total cartridge capacity of the drive is 2.16 million bytes. The actual figure will be smaller due to overhead of interrecord gaps and any software implemented error detection and correction methods used such as hamming codes, cyclic redundancy checks, or checksums. However, using 2.16 million bytes as an order of magnitude, this could for example hold 16,000 names and addresses of 133 characters each, 43,000 lines of source text averaging 50 characters per line, and so on. Very useful.■

A 2650 Single Board Computer



Jeff Roloff, whose company is called Central Data Company (POB 2484 Station A, Champaign IL 61820) has sent along this picture of a new $325 single board computer which uses the Signetics 2650 processor. This board includes an 80 character by 16 line video generator (upper case 64 character ASCII built-in) with a user programmable set of 64 characters (8 by 8 matrix) for nonstandard character sets. Also on the board are a 300 bps Kansas City standard cassette interface, a 1024 byte PROM monitor program, 2048 bytes of programmable memory of which 768 are available for programming with the balance (1280 bytes) dedicated to the display, one 8 bit input port and one 8 bit output port. The board contains sockets for an additional 3 K of PROM programming, which can include the firm's editor and assembler PROMs which cost $200 for a set. A BASIC package is also available on cassette tape for $20, as is a $20 tape version of the editor and assembler package. To use this computer, you'll need to supply a video monitor (or modified television), an ASCII keyboard, a power supply (5 V at 3 A) and probably some extra memory interfaced using DIP plug connectors on the board. The 2650, itself, is one of the unsung heroes of the microprocessor marketplace, a rather powerful machine with characteristics and addressing modes well worth investigating by the serious personal computing enthusiast.■

Interested in COSMAC Evaluation Boards?

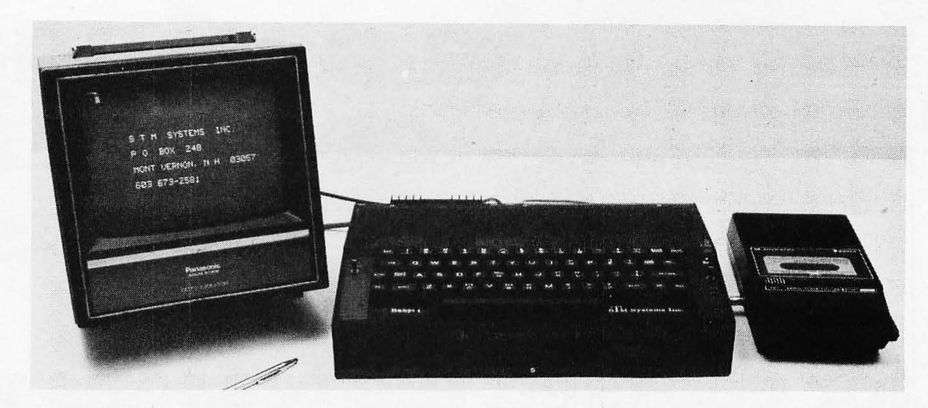
RCA Will Fix You Up with One of These Kits for $249 Direct or from Your Nearest Distributor . . .



This kit of components will allow the engineer or advanced homebrewer to put up a CDP1802 processor. It contains a printed circuit board, byte input and output ports, a terminal interface, systems software in ROM, and programmable memory for user programs. The board also features control logic and displays to aid in debugging of programs. To complete the system, you'll need a power supply of 5 V at 600 mA (or 10 V at 200 mA with 5 V at 400 mA for the LED displays) and a terminal with current loop (Teletype at 110 to bps) or RS-232 interface (110 to 1200 bps).

The board contains a total of 256 bytes of programmable memory using RCA's CDP1822 256 by 4 bit memory integrated circuits. Wired, but not stuffed, are positions which can accept 30 additional CDP1822 circuits for a total on board memory of 4 K bytes. The ROM system monitor consists of 512 bytes contained in one 512 by 8 bit CDP1832 read only memory. The program features include memory inspection and modification, initiation of user programs, and full interfaces to the communications terminal. The monitor's working memory is contained in a 32 byte CDP1824 memory part which is separate from the user memory. Since the whole system is CMOS, power requirements are quite low, allowing RCA to add the finesse of battery backup for memory contents. As a machine code level test and checkout device, the CDP18S020 board kit has controls in-

BABY!1 A COMPLETE AND FLEXIBLE



MICROCOMPUTER SYSTEM

LIGHT AND SMALL: BABY!1 weighs less than 10 pounds, is 14.75 inches wide, 10.625 inches deep and 3.625 inches high. Its attache case is slightly larger, leaving room for the powercord, video and cassette cables, cassette tapes, recorder, and paperwork.

POWERFUL MICROPROCESSOR: The 6502 processor has 13 addressing modes and 56 instructions. A single bus structure, engineered for high reliability and ease of use, gives it excellent versatility, expandability, and flexibility.

EASY PROGRAM DEVELOPMENT: Firmware (software in ROM) lets you enter and debug your programs through the keyboard and list them on the video monitor; lets you dump it to an audio cassette at 1200 baud for future use and exchange with other systems.

COMPLETE MONITOR IN FIRMWARE: No time-consuming loading of the monitor is required. Firmware controls the unit completely. Turn on the unit and its firmware will run the system until you enter a new program in RAM or plug in up to four other ROMS or PROMS. It controls the keyboard, video, and tape interfaces.

ULTRA RELIABLE KEYBOARD ENTRY: The unique keyboard with its 62 keys features ultra reliable solid state "No Switch" switching (no moving contacts), N-key roll over, tactile feel with a clicking sound when a key is depressed. Its MTBF (Mean Time Between Failures) is at least six years of continuous use, at least twice the life of an electromechanical keyboard.

HIGH-SPEED AND FLEXIBLE TAPE INTERFACE: The advanced 1200 baud cassette tape interface is under monitor control. You can write a program to simulate other speeds, techniques, and formats. Now you can read or write tapes for exchange on other systems! Any cassette recorder can be used. Two sets of cables are supplied, one for standard, the other for reverse polarity recorders. The system program provides a two to one ratio between ones and zeros, and loads at 1200 baud 4K bytes in less than 90 seconds.

INDUSTRY COMPATIBLE VIDEO OUTPUT: The video signal is EIA compatible 1 volt peak to peak. The flicker free DMA (Direct Memory Access) design displays 7 by 9 dot characters in 32 columns and 16 lines. Any page of memory may be displayed and edited. The 128 character set includes UPPER and lower case letters, and lower case γρεεκ. Four additional special characters control the DMA functions to hide or flash your message or to obtain 100 percent processor time.

EXPANDABLE RANDOM ACCESS MEMORY: Utilizing at least 4K bytes of the low power consumption and high-speed 2102 type chips, an additional 8K or 4K plus an expansion card can be added inside the system. The expansion card has two 50 pin connectors to enlarge the system.

MODULAR EXPANSION: With the expansion card, you may add up to 60K bytes of memory or connect and control any peripheral of your choice through the buffered bus, and through the four bidirectional interrupt driven parallel ports, the RS232 serial port, and the 20mA loop serial port. Connect printers, typewriters, tape drives, disks, floppy disks, additional processors, your railroad, or whatever else you may need.

COMPLETELY ASSEMBLED, TESTED, AND BURNED IN for 168 hours at high operating temperatures.

FULLY WARRANTED for 90 days on parts and labor.

MAINTENANCE CONTRACT available at the end of the warranty period on a yearly basis for your BABY!1 at a cost of less than 20 cents a day.

COMPLETE SOFTWARE SUPPLIED: A monitor stored in 1K ROM, and programs stored on a cassette include TINY BASIC, music, shooting stars, and a version of the powerful TECO™ text editor of Digital Equipment Corporation, and more.

TECO is a trademark of the Digital Equipment Corporation.

STM Systems Inc.

Specialists in Technology for Microprocessors

P.O. BOX 248
MONT VERNON, N.H. 03057
603-673-2581

BABY!1 BABY!1 BABY!1 BABY!1 BABY!1 BABY!1

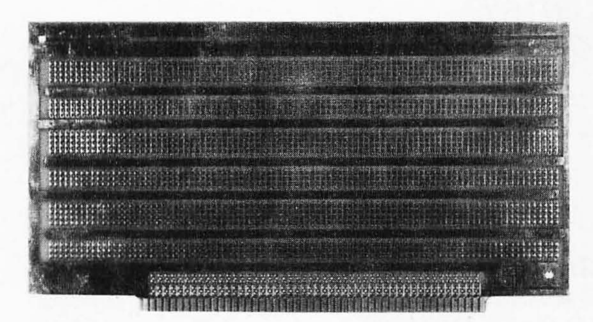
cluding a RESET button to initialize the system, continuous or single step operation, and displays of various information in LED devices. A prototyping area measuring 6 by 4 inches (15 by 10 cm) allows the user to adapt this system on a custom basis.

RCA Solid State is located at Route 202, Somerville NJ 08876, and as noted above, this kit is available for $249 direct or from your nearest RCA distributor. It's probably one of the best ways to take advantage of the 1802 instruction architecture in a homebrew computer project.■

Some Information on Refreshing Dynamic Memories Is Available from Texas Instruments

Texas Instruments has published a 12 page application summary bulletin entitled *Introduction to Refreshing TI 4 K Dynamic RAMs.* Bulletin MOSA3 introduces refresh principles and their implementation. It also compares static and dynamic RAMs in terms of speed, power consumption, refresh requirements, relative costs and power supply requirements. Various kinds of simple refresh circuitry are outlined and block diagrammed, including transparent, cycle steal and burst methods. Hackers and homebrewers looking for engineering information on use of dynamic memories should write to Texas Instruments Inc, Inquiry Answering Service, POB 5012, M/S 308, Dallas TX 75222. Attention: Bulletin MOSA3 Literature.■

Don Tarbell's New Prototyping Board . . .

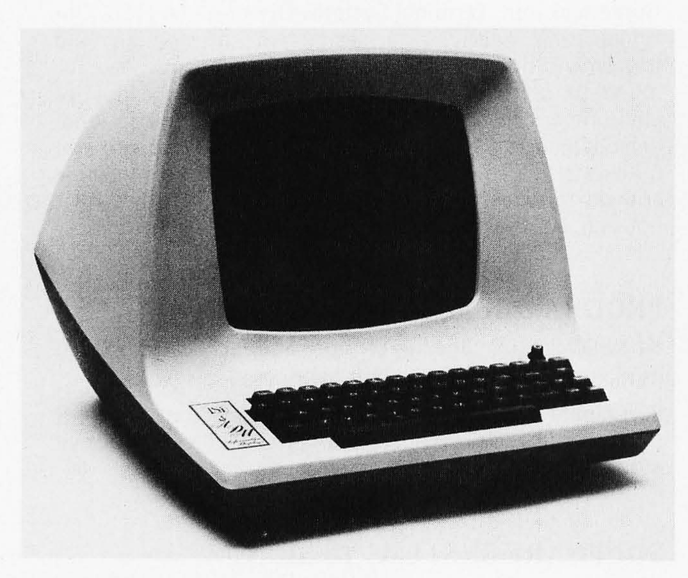


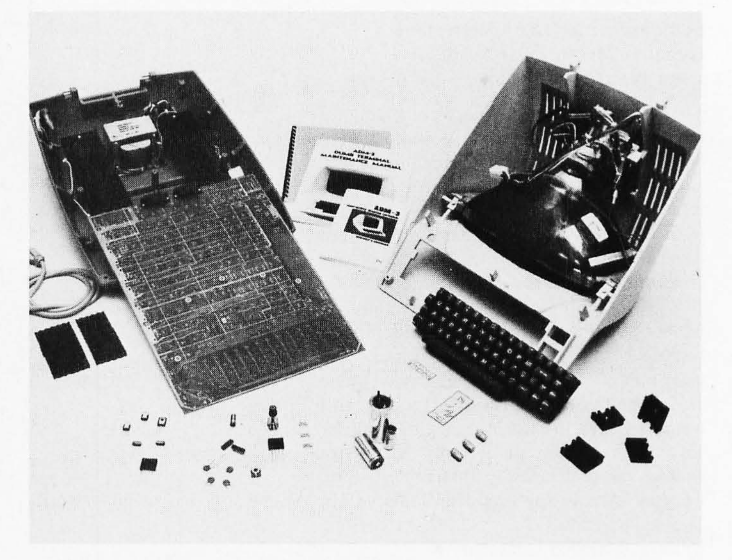
Here is yet another product for the Altair bus. Don Tarbell, Tarbell Electronics, 144 Miraleste Dr #106, Miraleste CA 90732, has just announced this new prototyping board for use with Altair and compatible computers such as the IMSAI 8080, Sol, Poly 88, etc. The board is designed with interleaved power and ground for the standard logic voltages, and has a position for a 5 V regulator (lower right side of the photo). The board is intended for use with point to point solder connections, wirewrap, or any combination. The board will accept 33 14 pin integrated sockets, or smaller counts of 16, 18, 24 and 40 pin sockets plugged into its pattern of predrilled holes. The price is $28 per board.■

synchro-sound enterprises

"THE COMPUTER PEOPLE"

" IN STOCK "
NEW LEAR SIEGLER ADM - 3A KIT
FULL ADDRESSABLE CURSOR





CHARACTER GENERATION
5 x 7 dot matrix.

DISPLAY FORMAT
Standard: 1920 characters, displayed in 24 lines of 80 characters per line.

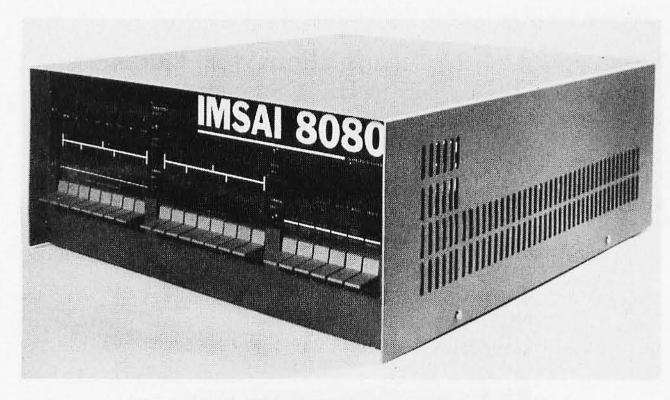
CHARACTER SET
Standard: 64 ASCII characters, displayed as upper case, plus punctuation and control.

COMMUNICATIONS RATES
75, 110, 150, 300, 600, 1200, 1800, 2400, 4800, 9600, 19,200 baud (switch selectable).

COMPUTER INTERFACES
EIA standard RS232C and 20 mA current-loop (switch selectable).

DATA ENTRY
New data enters on bottom line of screen; line feed causes upward scrolling of entire display with top-of-page overflow. Automatic new line switch selectable, end-of-line audible tone.

ADM-3A KIT . . . . $ 895.00
ADM-3A ASSEMBLED 1099.95
LOWER CASE OPTION 89.00

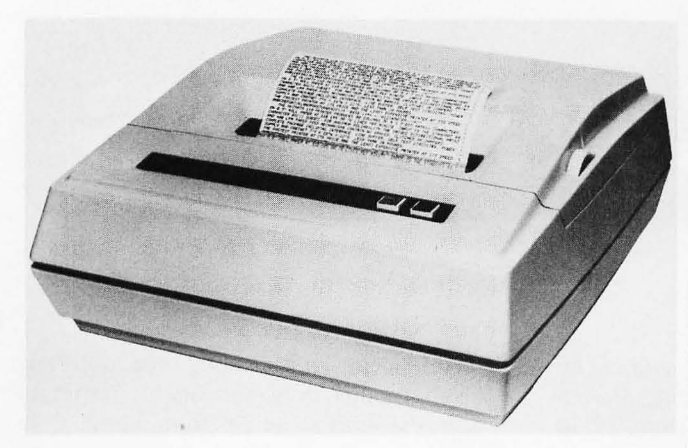


IMSAI 8080 MICROCOMPUTER
POWERFUL • EASY TO USE • LOW COST

$619.95/22 SLOT MOTHER BOARD
849.95/WITH Z-80 CPU

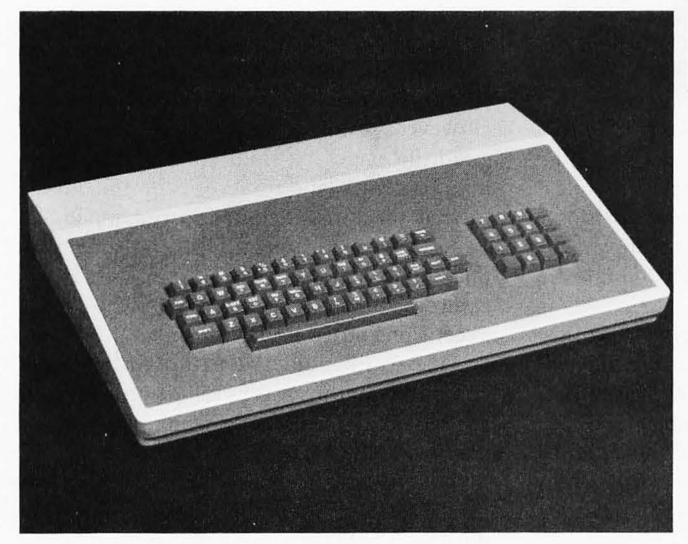
FOLLOWING ITEMS IN STOCK TDL, OAE, PROCESSOR TECH., SWTP, APPLE, HAYDEN, TARBELL, IMSAI, LEAR SIEGLER, OKIDATA, DEC, JAVELIN, TELETYPE ASR-33, TRW.

SPECIAL 9" JAVELIN HIGH RESOLUTION VIDEO MONITOR $159.95



($1099.00)
OKIDATA MODEL 110
110 CPS DOT MATRIX LINE PRINTER

FRICTION FEED*	$1099.00
TRACTOR FEED	1229.00
RS 232C SERIAL INTERFACE	250.00



* *

MOST ITEMS IN STOCK FOR SAME DAY SHIPPING. FULL MODERN REPAIR FACILITIES ON PREMISES—WE SERVICE WHAT WE SELL.

* *

synchro-sound enterprises

193-25 Jamaica Ave., Jamaica, NY 11423
Phone (212) 468-7067

HOURS 9 – 4 DAILY + SATURDAY
BANKAMERICARD MASTER CHARGE
VISIT OUR NEW SHOWROOM
WORKING UNITS ON DISPLAY

DECwriter II



$1769.95

Features
132 column printing
30 CPS
Full Keyboard
Tractor Feed

NEW COMMERCIAL QUALITY KEYBOARD

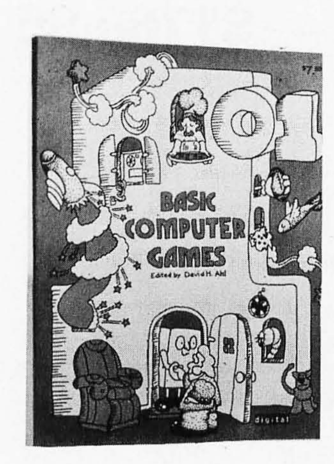
The Model SS-1 Communications Terminal is a non-contact capacitive keyboard with a guaranteed life of over 100,000,000 operations.

FEATURES

- **MOS/LSI Encoder** with high reliability and low power consumption
- **n Key Roll Over** which eliminates operator error and increases thruput
- **Hysteresis** for tease proof operation
- **Solid State** performance at mechanical switch prices
- **Tactile Feedback** at the operate point
- **ASR-33 Array** with four mode encoding

KEYBOARD KIT	$ 79.95
ENCLOSURE (WITH NUMERIC CUTOUT)	39.95
NUMERIC PAD	29.95
COMPLETE KIT (with pad)	139.95
ASSEMBLED UNIT (with pad)	189.95

Book Reviews



101 BASIC Computer Games *edited by David H Ahl, Creative Computing, Morristown NJ 07960, 248 pages softbound. $7.50 plus postage.*

101 BASIC Computer Games, edited by David H Ahl, is not only the first collection of games all in BASIC, but a uniquely educational book which provides both a complete listing and description of every game along with a sample program for each.

As Ahl points out in his book, educators generally agree that games are highly motivational and promote learning by discovery. What better way is there to learn about Newton's second law than by simulating an Apollo lunar landing in ROCKET? Or to learn about logic by playing BAGLES? You can even increase your vocabulary while playing SYNONM or improve your writing skills in BUZZWD by learning how to compose computer speeches with the latest buzzwords.

For those interested in more exotic games, there's CHEMST, in which the player tries to dilute the fictitious kryptocyanic acid; CHOMP, which involves eating a cookie while trying to avoid the poison piece; and HELLO, in which the computer dispenses advice on such problems as sex, health, money or a job.

Computer enthusiasts with a sense of humor will find many entertaining games in the book with such challenging objectives as delivering pizzas successfully, PIZZA; doing a silly profile plot of an ugly woman, UGLY; and finding the happy hurkle beast hiding in a 10 by 10 matrix, HURKLE.

The names alone of many of the games are intriguing enough to invite further investigation. FIPFOP, SPLAT and ZOOP are sure to send hobbyists running to their computers. FIPFOP is a solitaire logic game dealing with changing a row of Xs to Os. SPLAT involves opening a parachute at the last possible moment. ZOOP, otherwise known as the BASIC programmer's nightmare, is designed to imitate the system commands of a BASIC compiler, except that it gives totally meaningless and frustrating results.

Ahl spent considerable time collecting this potpourri of games on his travels to various schools as well as from submittals in response to an advertisement. Game authors range from seventh graders in California to PhDs in England.

The games run the gamut from extremely simple to more complex; but most require no special knowledge. To solve the game categorization dilemma, Ahl has simply listed the games in alphabetical order. But in the appendices, he has outlined some family groupings, such as logic, plotting and matrix games.

A BASIC speaking computer is the only equipment needed to play any of the games. However, Ahl suggests that a grid or quadrille paper be used to play four of the matrix games and that one of the supplemental diagrams included in the appendices be used when playing QUBIC, a game of tic tac toe in a 4 by 4 by 4 cube. Most of the games also run in standard BASIC with any exceptions noted under the game title.

Due to the addictive nature of the games, computer enthusiasts should be reminded not to skip meals or sleeping in favor of playing ANIMAL or FOOTBL.

101 BASIC Computer Games is available in a 248 page softbound edition for $7.50 plus $.75 postage from Creative Computing, POB 789-M, Morristown NJ 07960, and from BITS Inc.

Linda Blocki
1706 Silver SE, Apt 27
Albuquerque NM 87106■

TV Typewriter Cookbook *by Don Lancaster. Howard W Sams & Co, Indianapolis IN 46268, 1976. $9.95.*

Don Lancaster's *TV Typewriter Cookbook* provides comprehensive coverage, not only of the problems of generating character displays on a home TV set, but of a host of related topics as well – memories, keyboards, cassette recording methods, modems, and hard copy devices. His book is one of the most valuable yet to appear for the electronics and computer hobbyist.



Some of the chapters in this book appeared as articles in the earliest issues of BYTE (September 1975 – January 1976, March 1976), and several of them won awards in our BOMB (BYTE's Ongoing Monitor Box) contest. If you've been looking at some of the TV interface products on the market and have wondered how they work; or if you'd like to know more about programmable read only memories, UARTs, keyboard encoders, bus transceivers and other fancy ICs, this is the book for you.

Don is one of the few writers who not only presents circuits, but explains what's really going on in them, and what design alternatives exist. The book starts with basics, like how TV scanning works, and how data is represented in the ASCII, Baudot and Selectric codes. Successive chapters describe memories (both random access and read only) and buses; timing, sync signals and composite video for display generation; and cursor and update circuits. Next Don digresses for two valuable chapters to deal with keyboards and encoders, serial data handling, UARTs, cassette recording, modems, and telephone characteristics. Then his attention returns to the main TV typewriter problem as he describes methods for interfacing to the TV itself (both direct video and RF entry), color techniques, and graphics. The book concludes with a discussion of hardcopy alternatives, including some useful, if sketchy, information on the possibility of converting a Selectric typewriter for computer use.

This book has been out for some time, and should be available through many bookstores and retail computer outlets (also through BITS). Many of our readers may already have this book. Those who haven't seen it would do well to check it out.

Software Tools *by Brian W Kernighan and P J Plauger, Addison-Wesley Publishing, Reading MA 01867, 1976. $8.95.*

This book could be the jumping off point for creation of a great deal of useful software for personal computers. Its great strength is that it contains well written, thoroughly documented, tested and working programs for many of the most desired home computer applications: a very powerful text editor, a text formatter for document preparation, a macro processor, and a SNOBOL like string pattern matching program. Also included are general purpose sorting programs, a file management package, and a variety of useful utility routines.

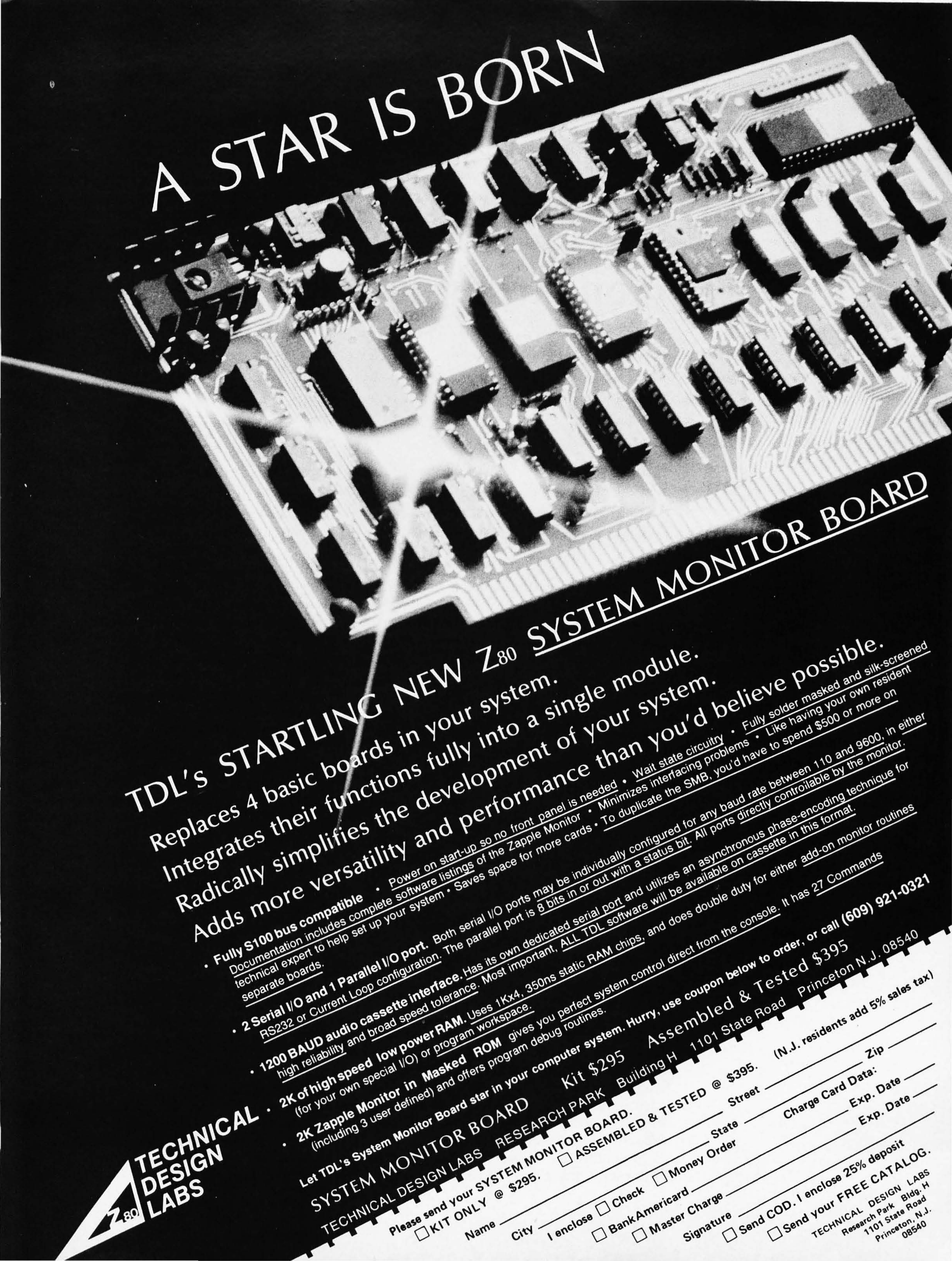
The purpose of this book, which was written by the authors of the highly recommended *Elements of Programming Style* (see BYTE volume 1 number 1), is to teach good programming practice by presenting a series of practical programs, each of which provides lessons in program design and implementation. All of the programs are general purpose tools which are used by the authors in their everyday work; in fact, many of them were used to help prepare the text of this book. The programs form a comprehensive set which is designed to be as easy as possible to use. The code is interleaved with explanatory text which describes, not only how each program works, but why it was designed and implemented that way.

The programs are written in a structured version of FORTRAN, called RATFOR (for "rational FORTRAN"). RATFOR simply provides control structures such as nested IFs and WHILE-DO and REPEAT-UNTIL looping constructs. The remaining statements utilize a subset of ANSI FORTRAN features that should be found in nearly all FORTRAN implementations. The last chapter of *Software Tools* presents, naturally enough, a RATFOR-to-FORTRAN translator.

There are two ways to use this book to develop software for personal computers: First, the RATFOR source code could be hand translated to the assembly language of one or more of the popular microprocessors. This would be a lot less work than designing such tools from scratch, and should make debugging a far less formidable job. Second, a "Tiny FORTRAN" compiler or interpreter could be developed which would make it possible to mechanically translate or interpret the programs on a very small machine. Only the subset of FORTRAN required by RATFOR would have to be implemented. The only drawback to this approach is that some of the programs presume the existence of a "nice" operating system (the authors use UNIX on the PDP-11), which is sadly lacking on most existing personal computers.

To assist in the second approach, the RATFOR source programs in machine readable form can be obtained from the publisher. Write to Linda Banks, c/o Addison-Wesley Publishing Company, Reading MA 01867, (617) 944-3700. The cost is just $25 plus $1.67 postage for an IBM/360 compatible 9 track, 800 bpi, odd parity, EBCDIC coded magnetic tape. We'd like to hear from all the system hackers out there who pursue this approach.

Daniel Fylstra
Hamilton Hall C-23
Harvard Business School
Boston MA 02163■



A STAR IS BORN
TDL's STARTLING NEW Z80 SYSTEM MONITOR BOARD
Replaces 4 basic boards in your system.
Integrates their functions fully into a single module.
Radically simplifies the development of your system.
Adds more versatility and performance than you'd believe possible.
• Fully S100 bus compatible • Power on start-up so no front panel is needed • Wait state circuitry • Fully solder masked and silk-screened Documentation includes complete software listings of the Zapple Monitor • Minimizes interfacing problems • Like having your own resident technical expert to help set up your system • Saves space for more cards • To duplicate the SMB, you'd have to spend $500 or more on separate boards.
• 2 Serial I/O and 1 Parallel I/O port. Both serial I/O ports may be individually configured for any baud rate between 110 and 9600, in either RS232 or Current Loop configuration. The parallel port is 8 bits in or out with a status bit. All ports directly controllable by the monitor.
• 1200 BAUD audio cassette interface. Has its own dedicated serial port and utilizes an asynchronous phase-encoding technique for high reliability and broad speed tolerance. Most important, ALL TDL software will be available on cassette in this format.
• 2K of high speed low power RAM. Uses 1Kx4, 350ns static RAM chips, and does double duty for either add-on monitor routines (for your own special I/O) or program workspace.
• 2K Zapple Monitor in Masked ROM gives you perfect system control direct from the console. It has 27 Commands (including 3 user defined) and offers program debug routines.
Let TDL's System Monitor Board star in your computer system. Hurry, use coupon below to order, or call (609) 921-0321
SYSTEM MONITOR BOARD Kit $295 Assembled & Tested $395
TECHNICAL DESIGN LABS RESEARCH PARK Building H 1101 State Road Princeton N.J. 08540
Please send your SYSTEM MONITOR BOARD.
☐ KIT ONLY @ $295. ☐ ASSEMBLED & TESTED @ $395. (N.J. residents add 5% sales tax)
Name
Street
City
State
Zip
I enclose ☐ Check ☐ Money Order
☐ BankAmericard
Charge Card Data:
Exp. Date
☐ Master Charge
Exp. Date
Signature
☐ Send COD. I enclose 25% deposit
☐ Send your FREE CATALOG.
TECHNICAL DESIGN LABS Research Park Bldg. H 1101 State Road Princeton, N.J. 08540
TECHNICAL DESIGN LABS
Z80

Why Aren't There Any Altairs on Arcturus II?

Henry Melton
7307 E Riverside Dr
Lot #13
Austin TX 78741

As in all prophecy, only the correct predictions are remembered, and even those aren't outstandingly accurate . . . compare the actual nature of the first moon flights with the predictions.

I have been writing science fiction for some time now, and I've been reading it forever, so it doesn't strike me as being very odd that I'm now into computers. If you mention your computer to a friend, you're likely to get a HAL joke back. It's an automatic association: Computers and science fiction just seem to go together. It's logical too. For as long as I can remember, the science fiction stories I read had ideas. Spaceflight, nuclear power, robots and a hundred other marvels were accepted features of those adventures. A good science fiction yarn would drag me off to some strange time and place where there would be far too much going on for me to ponder over the particular space drive involved, or to puzzle about the program that might be running beneath the robot's polished skull. But when I got back to Earth, how things had changed. So many marvels seemed so possible in what had been such a drab mundane world.

Well, the moonflights and the nuclear power plants and the Viking Lander robots happened. Everyone speaks of today as being a science fiction world. And, in a sense, the wonders of today were spelled out years ago in the books and magazines of science fiction. Science fiction writers are treated as modern day prophets and people expect them to have a handle on the future.

In another sense, however, a closer look at some of these predictions shows a less flattering picture. Only the correct predictions are remembered, and even those weren't outstandingly accurate. Nobody predicted such a complicated first moon flight, with command modules and orbiting stages, descent and ascent staged landers. Few stories gave the first moonship an onboard computer. None included television cameras.

But no one worries too much about that. After all, one person sitting at a typewriter can't really compete with a team of engineers in working out the best way to put a man on the moon. A science fiction writer isn't a true prophet; all he or she can do is choose a possible future to write about. Just as the human race, by its actions, chooses a possible future in which to live. If a writer knows humans well enough, he/she can second guess the race and look like a pretty fair prophet. Such is the way of the game.

Looking back at the hundred million words of science fiction I have read to date, I think I have gotten a good return. Future shock holds no terror for science fiction readers. No new technological marvel can sneak up and go boo. The science fiction reader has seen it all before.

Or, rather, almost all of it.

There does seem to be a big black gap in this flood of prophecy. The science fiction reader who is familiar with the common picture of a computer as portrayed in science fiction is due for a big shock when he/she runs up against the powerful little critter called a microprocessor. It's here, now, with a potential to open up society to private initiative in a manner almost unprecedented, and nobody wrote a story predicting it.



That is frightening in itself. What is wrong with science fiction that something so technological as a revolution in computers could go unheralded? Are science fiction writers losing their touch? Is technology developing too fast for them to keep up?

An overview of modern science fiction, however, indicates that this gap seems to be a strange, localized thing. In other areas of human knowledge, science fiction is still riding high on the far edge of the barely possible. Interstellar ramjets, galactic core explosions, Kerr black holes, gene grafting, just about any conjecture of physics, cosmology, biology or whatever, is likely to be found in a modern science fiction story. Particularly in physics, some stories have come out in the science fiction magazines before the original research on which they were based has even made it to the professional journals. Science fiction can still make its claim to be the literature of ideas.

But, Alfred Bester's excellent novel, *The Computer Connection,* has a room sized malevolent computer trying to take over the world. Roger Zelazny's "Home Is The Hangman," which has just won the Hugo award for the best science fiction novella of 1975, has the main character hunting down a possibly murderous robot, before the robot can find his creators. Isaac Asimov's "The Bicentennial Man" follows a robot,

Future shock holds no terror for science fiction readers.

Yet, this gap in the literature is a small one . . . science fiction can still make its claim to be the literature of ideas when it talks about the social implications of black holes, the people dynamics of a runaway relativistic acceleration in an interstellar ramjet, or what to do about a murderous robot.

If a writer knows humans well enough, he/she can second guess the human race and look like a pretty fair prophet. Such is the way of the game.

The personal computer is here now, with a potential to open up society to private initiative in a manner almost unprecedented in history, and nobody wrote a story predicting it. Are science fiction writers losing their crystal balls?

originally designed to be a butler, in his lifelong quest to become a human being.

Now, these are good stories, some of the best of the past couple of years. But the computers and robots in them are no more sophisticated than those in science fiction stories of twenty years ago.

Science fiction has had its stereotypes. The two computer types constantly used are either the huge device, something that would have been a good university computer back in the sixties, except for its disconcerting tendency to chuckle evilly when no one is looking, and the robot. The robot, moreover, usually has some kind of magical (ie: Asimov's positronic) brain that is really nothing more than technological handwaving on the author's part to let him have a human brain sized computer inside a human looking robot body.

The real shame is that these stereotypes haven't changed in the twenty years that they have been used. A reader of BYTE can sit down with one of today's science fiction books and rest his/her reading arm on top of a logic machine considerably more sophisticated than anything he/she is likely to encounter in the pages of the book. It is a shame, if for no other reason than some otherwise good stories are going to be unpalatable to a lot of personal computer owners. This mental gap in science fiction writing really shows up in some places. Can you imagine what Spock would have done if there had been a microprocessor in his tricorder as there should have been?

The more one thinks about it, the clearer it becomes that science fiction writers are really behind schedule when it comes to computers; actually behind the times, when they should be well ahead. It is a blank spot in science fiction, and it has been there for years. It might have stayed blank for many more years if it hadn't been for the invention of the microprocessor. The only people who can notice that blank spot are those who know better. In other words, only you and I and the people we talk to.

But there are signs that the writers are waking up. I think the pocket calculators caught us all off guard. Poof! There it was, a gadget with major sociological implications in everybody's pocket, and nobody had really predicted it. I can remember only one old story that used pocket calculators, and even then, they were finely machined, motorized slide rules. The story was good, though. It explored what a world would be like if everybody had forgotten that math could be done in one's head. If the story had come out only five years ago, it would have been hailed as a prime example of the predictive value of science fiction. *But no* modern story had even considered the concept.

That was a shock for the writers; and signs that this stagnation is breaking up are starting to appear in print. But why was that mental block there at all, and why has it persisted for so long? Was it just a fluke?

I don't think so. I'll tell you why.

A quick look at all of the computer stories that have come out since computers and science fiction writers discovered each other back in the forties will show one dominant theme: artificial intelligence.

A writer of fiction stories has to be concerned with people. Even a gadget story is only a story inasmuch as it affects people. For a writer to write anything worth reading, he/she has to consider the characters above all else. When the intelligent computer first appeared on the science fiction scene, everybody took the concept to heart.

The intelligent computer made a beautiful character, from the storyteller's point of view. The full range of personalities, from purely logical to insanely demonical, were all available for use, and all essentially believable. Believability is a prime quality in science fiction. It has to be there if the story is going to be any good. And anyone can believe a computer with a bad case of misprogramming.

And so the computer became part of the stable of characters available for a science fiction writer, right up there with human beings, mythological creatures, and aliens from the planet ______. *[The symbolic name of your favorite mythical planet is a parameter to be supplied by the user of a science fiction writing program.]* A computer was a character. The concept became so fixed that writers forgot that real computers exist because they are beautiful tools. And while science fiction writers told their tales of pensive robots and planet killing Berserker machines from some war lost long ago and light centuries away, the macrocomputers begat minicomputers, and minis begat micros. Out of the same labs came the

pocket calculators, which jumped out on the market and gave us all a little taste of future shock.

The shock has done some good. Some of the better writers are already into the swing of things. Take two examples: You must read *The Mote In God's Eye* by Larry Niven and Jerry Pournelle, and *Imperial Earth* by Arthur C Clarke. In both of these novels everyone has a pocket computer. These gadgets are pocket sized with large memories and very easy to program. They can store text, graphics, and sound, occasionally tying into larger computers by radio, thus serving as diary, library, calculator, and who knows what else. This pocket computer is such a logical development that you can bet that other writers will pick up on the idea. Here is a beautiful tool in science fiction, and not a hint of an evil chuckle out of it.

Here is another example: *Shockwave Rider,* by John Brunner, extrapolates a world where any computer can be reached and programmed from any telephone. (This is an extrapolation?) In this future world, the programming genius with the right password is like the one eyed man in the country of the blind. What can be forbidden to the man who can change his identity, profession, and financial status, with just a couple of hours tapping touchtones on his phone? Shades of stories we've read about in *Computerworld*! In this story, Brunner introduced the concept of a tapeworm to the science fiction readership.

A tapeworm is a software life form. It lives in memory space, eating processor time. And they exist. I've seen them. So have you if you've watched a program blow up, using a video display as a window into your memory space. I've watched about five naturally occuring species in a friend's 8080. If you are doubtful that they are alive, go back and reread the definition of life and think again.

Exciting things are starting to happen as science fiction takes another look at computers and what they can do. But only you can determine how fast these stories can come out. You can't wake up a writer if you haven't met him/her. So put on your BYTE tee-shirt and go to the next science fiction convention and meet the people. Talk about your machines.

The computer as a beautiful character will never go away; you can be sure of that. But the beautiful tool is here, and all it takes is for science fiction writers to realize it. Then, finally, you might see an Altair on Arcturus II.■

When the intelligent computer concept first appeared on the science fiction scene, everybody took the concept to heart. Aha, here's another type of character!

Definition: A tapeworm is a software life form. It lives in memory space, eating processor time. The etymological roots of its name lie in the theory of generalized computability, ie: the theory of Turing machines.

Navigation with Mini-O

Part 3, Software

Richard J Salter
Avionics Engineering Ctr
Ohio University
Athens OH 45701

The previous two articles on Omega Navigation and the Mini-O design by Ralph W Burhans */February 1977 BYTE, page 62; March 1977 BYTE, page 70/* have presented the basics of the Omega system and the hardware required for a minimal Omega receiver interface to microprocessor systems. In this part we will present the software details for phase-tracking loop filters and methods for output of the data as filtered lines of position (LOPs) for selected station pairs. This type of data output would be directly usable in the simpler forms of marine navigation receivers with a digital to analog converter driving a low cost Rustrak type of chart recorder.

The original concept of the Mini-O Omega receiver by Ralph W Burhans has been augmented by the contributions of a number of individuals from the Ohio University Avionics Engineering Center, Omega project team. These include Dr Robert W Lilley who first suggested the concept of the Memory Aided Phase Locked Loop (see reference 1) and also originated the basic idea of the software based Omega sensor processor in the early stages. Kent Chamberlin perfected the Memory Aided Phase Locked Loop in practice in 1974 with an all-TTL hardware version described in reference 2. The design has subsequently been converted to CMOS logic and implemented in a number of prototype receivers in our lab. These CMOS digital hardware sensors have been prepared by Ohio University under the sponsorship of the NASA Langley Research Center program in air transportation for eventual use by the general aviation community.

The basic goals of this NASA sponsored program have been to provide the general aviation pilot with some alternative navigation aids for use on a world wide basis from the Omega VLF signals. The feasibility of low cost Omega data sensor methods for airborne use has been demonstrated in a number of Ohio University test flights. The program is now continuing with the development of navigation software to provide the pilot or navigator with more direct data readout in the form of course deviation and indications of distance to go.

An outgrowth of the program's airborne Omega receiver research is the Mini-O system.

The Mini-O interface, operating at a 40 Hz sampling rate, is a spin-off of an earlier investigation using a 100 Hz sample rate and a more expensive temperature compensated crystal oscillator clock operating at 2.611200 MHz (10200 X 2^8 Hz). (See reference 3). In this investigation, both ground and in-flight measurements of actual Omega phase data were collected. The data base was then used as the input for a software (FORTRAN language) simulation of the Omega sensor processor functions. This simulation analysis, performed on actual off-the-air Omega data, proved the feasibility of software methods of manipulating Omega data at a low interrupt rate.

Digital Phase Locked Loops

A very simple form of digital phase locked loop has been presented in part 2 of this series where a pair of exclusive OR gates provide a phase detector for comparing the filtered 40 Hz signal obtained from a down counter, with the phase of the signals obtained from the type D flip flop used as a mixer (see figure 7 of part 2). This is what we call an increment-decrement type of binary phase locked loop where the 40 Hz output is being continuously generated from 2^{15} Hz ÷ 2^9 and 2^{14} Hz ÷ 2^9 depending on where the incoming noisy edges are with respect to the divide by 2^9 counter output. This provides a very effective low pass filter for the 40 Hz signal with a lock up time of a few cycles which is comparable to the 14 Hz bandwidth of the analog RF front end filtering. For the interested reader, a thorough treatment of digital phase locked

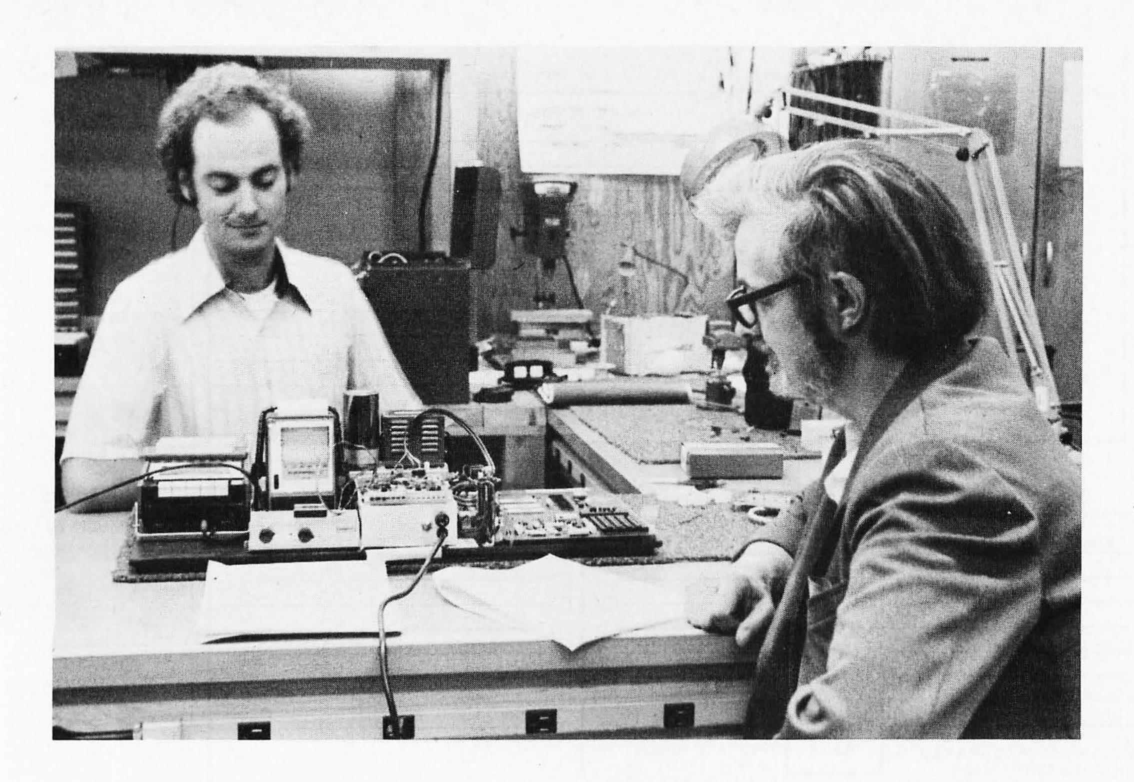


Photo 1: Authors Richard Salter (left) and Ralph Burhans (right) shown with the Mini-O receiver equipment driven by a KIM-1 processor seen at the right.

loops is available in reference 4.

The Memory Aided Phase Locked Loop is a more complex digital filtering system. In the hardware version, the output of a temperature compensated crystal oscillator operating at the Omega reference frequency of $2^6 \times 10.2$ kHz is divided in a 6 bit binary divider chain, and the outputs of all six stages are compared with the six most significant bits of an up down counter which represents the current Omega Phase estimate. This comparison yields a pulse at the point in time of coincidence of the two 6 bit numbers. The locked pulse is in turn fed back to the phase detector which compares the time of arrival of the locked pulse to the time of arrival of the next incoming Omega zero crossing. This phase detector is similar to the one previously discussed, and it outputs an increment or decrement signal to the loop counter.

The memory aiding feature allows a single loop (in software here) to track the signal phase in all eight Omega time slots by periodically (ie: every ten seconds) loading the phase estimate from a particular time slot back into the loop counter when it is again time to begin tracking that particular time slot's signal. In this time-multiplexed fashion a single loop and eight words of memory can do the work of eight separate phase locked loops.

Using the memory aided phase locked loop, a quite narrow bandwidth digital filter for the phase information produced by the Omega front end can be achieved. Ten bits of up down counter filtering before the 6 bit comparators provides a bandwidth of about 1/10th of an Omega cycle. This only allows the tracking loop to move about 30° from the previous phase value during a single time slot (using a tracking window of 625 ms per time slot).

Selected pairs of the 6 bit numbers obtained from the memory aided phase locked loop after each time slot can be subtracted, thereby generating Omega lines of position values. The subtraction as performed in hardware uses parallel arithmetic adders as two's complement subtractors and appropriate latches to store the data for the selected stations.

REFERENCES

1. Lilley, R W, "Binary Processing and Display Concepts for Low-Cost Omega Receivers," *Journal ION*, Volume 22, Number 3, Fall 1975.

2. Chamberlin, K, "Binary Phase Locked Loops for Omega Receivers," *Proceedings of the Second Omega Symposium ION*, Washington DC, November 1974.

3. Lilley, R W, and R J Salter, "Simulation Analysis of a Microcomputer-Based Low-Cost Omega Navigation System," *Proceedings of the National Aerospace Symposium of the ION*, Warminster PA, April 1976.

4. Reddy, C P, and S C Gupta, "A Class of All Digital Phase Locked Loops: Modeling and Analysis," *IEEE Transactions on Industrial Electronics and Control Instrumentation*, November 1973.

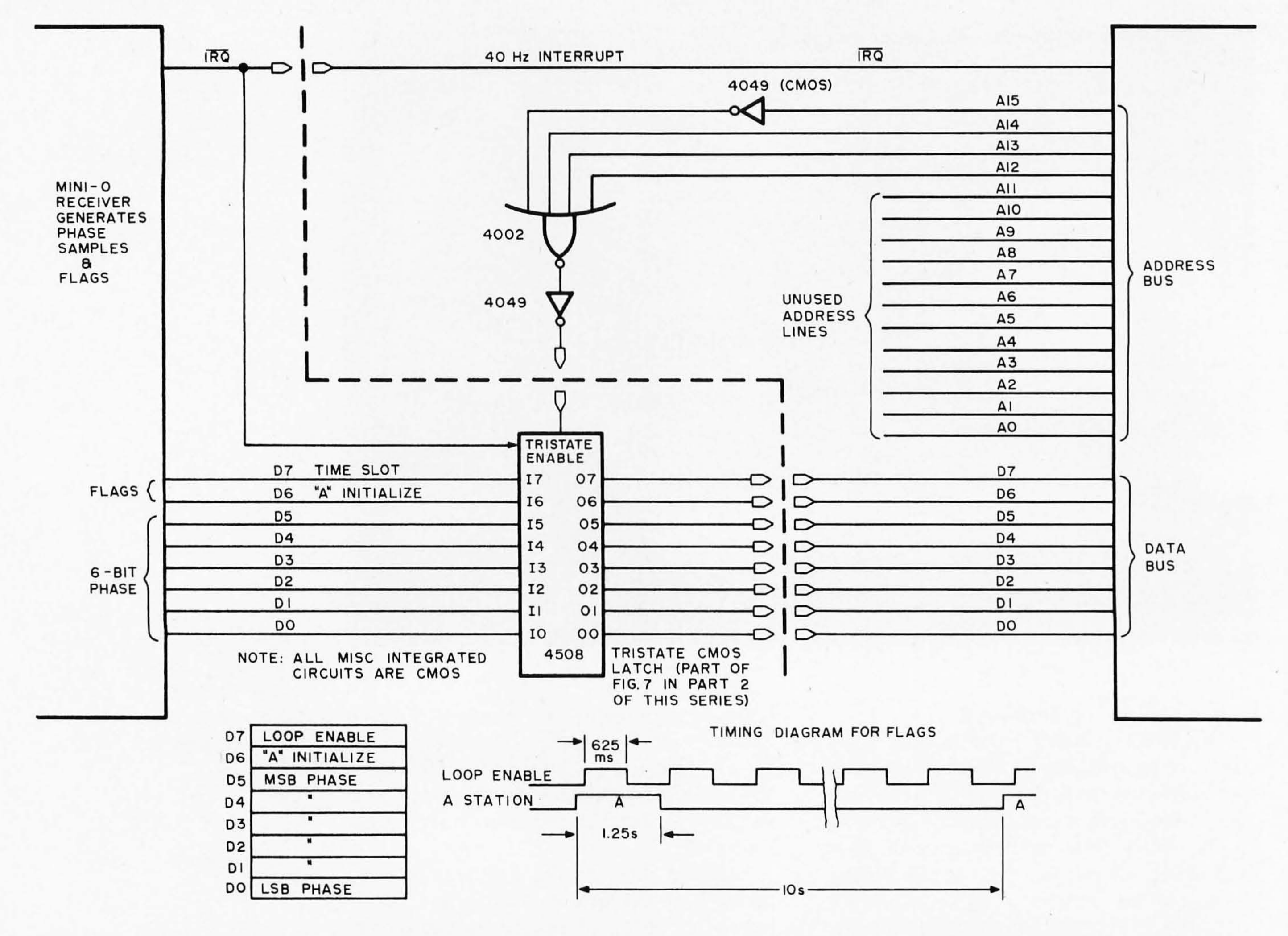


Figure 1: Using memory mapped IO ports. In small systems, where little memory address space is committed to programmable memory resources, it is possible to use the high order address lines to partially decode IO ports. This technique is applied here to show how the KIM or JOLT computers address the Mini-O receiver's eight bits of input data. There is one hitch with this form of IO address decoding: Since the low order address bits A0 to A11 are not used, all addresses from hexadecimal 8000 to 8FFF are equivalent in this particular case. In the logic of figure 3, a similar ambiguity of addressing finds hexadecimal 9000 to 9FFF treated identically as the output analog value latch.

With the phase locked concept thus described as a primer for the software tracking system to be presented, the remainder of this article concentrates upon the specifics of the microprocessor interrupt IO and the Omega sensor processor software originally used with a JOLT Microprocessor System. The JOLT system is a product of Microcomputer Associates and is based on the MOS Technology 6502 CPU chip. The basic JOLT system, incorporating the 6502 CPU, 6530-004 Teletype Monitor ROM, and ASR-33 and RS232 Teletype interface hardware (also a PIA which is not used in this application), is available on a single 4-1/2 X 7 inch printed circuit board for $150 in kit form. Photo 1 shows a KIM-1 processor by MOS Technology; the programs shown in listings 1 and 2 will work with any 6502 processor based system by using appropriate monitor service routine calls in place of JOLT's TIM monitor calls.

Memory Mapped IO

In the name of system simplicity and low cost, the data input and output decoding technique known as memory mapped IO is sometimes used. As mentioned previously, the JOLT system includes a peripheral interface adapter chip (PIA) which can be used for IO. However, in a small dedicated system with only one or two interrupts to service, the generality and sophistication provided by a PIA can become an expensive luxury — especially when one realizes the simplicity inherent in a memory mapped input latch.

The memory mapped input circuitry is illustrated in figure 1. As used with the microcomputer interface described by Ralph

Burhans last month (see figure 7 in part 2 of this series), the memory mapped input hardware consists of a NOR gate, inverter and a tristate latch. The NOR gate and inverter are used to decode the four most significant address bits, and the decoder output enables the Omega data onto the data bus by enabling the tristate latch when the Omega data port is addressed by interrupt service routine software. When the microcomputer receives the 40 Hz $\overline{IRQ}$ from the Mini-O interface, the service routine needs only to say "load accumulator with data from memory location 8XXX" to accept the new Omega phase data. Note that since the $\overline{IRQ}$ is a level sensitive input the $\overline{IRQ}$ signal must be conditioned into a short pulse so that it is not serviced more than once. It is necessary to add the circuitry in figure 2 to figure 7 of part 2 to obtain this $\overline{IRQ}$ pulse.

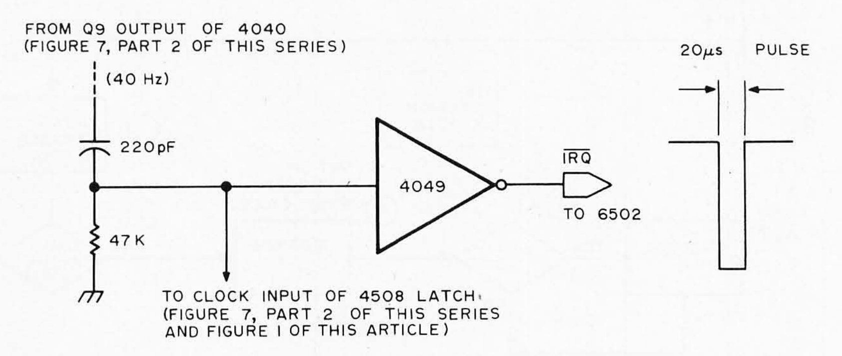


Figure 2: Since the IRQ signal of the microprocessor is a level sensitive one, it is necessary to shorten the pulse out of the Omega receiver shown in last month's article. A method appropriate for noncritical applications is use of an RC time delay. This RC delay developed by a 200 pF capacitor and 47 k resistor provides a pulse width of approximately 20 μs on the IRQ line. In applications where critical timing is involved, this method is not recommended.

Since only the four most significant address bits are decoded in the memory mapped input port, any attempt to address the 4 K locations encompassed by 8XXX will result in enabling the Omega input latch. In effect, 4 K bytes of potential memory space have been delegated to this one input port; however, since the 16 bit address bus permits addressing up to 64 K bytes, this apparent liberal use of memory space is of

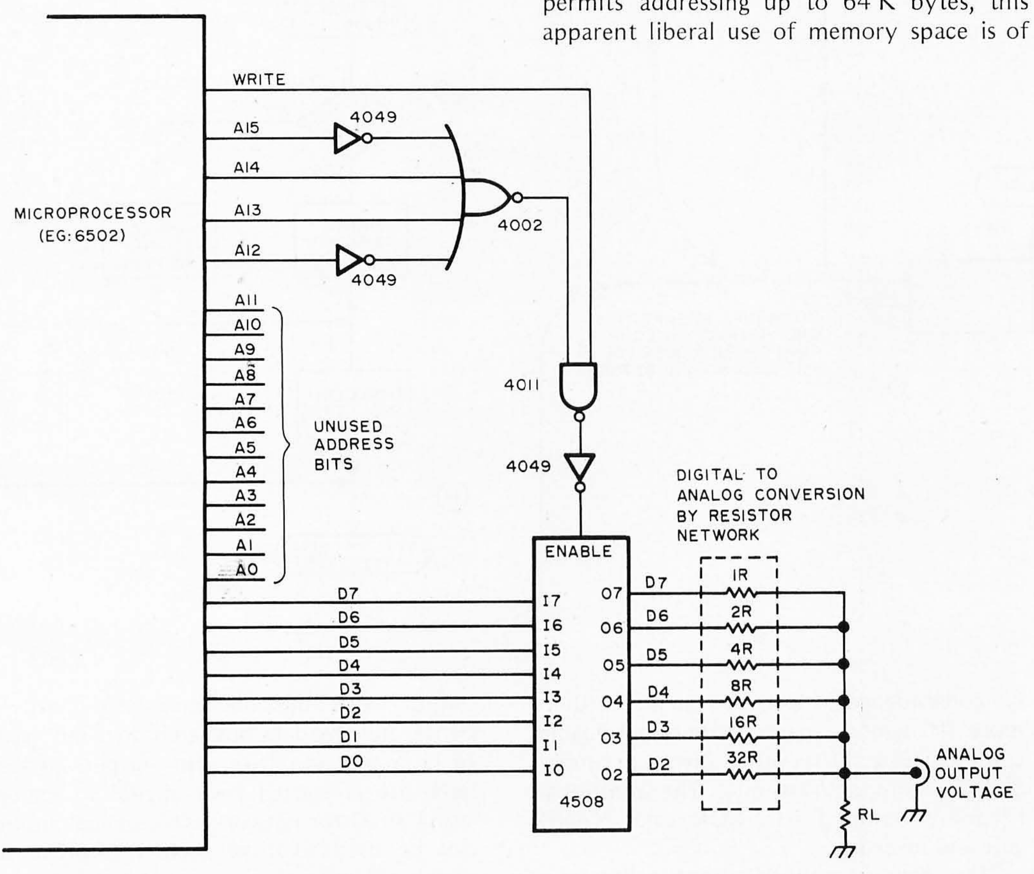


Figure 3: An output interface mapped to memory locations 9000 to 9FFF (hexadecimal). This is a simple 8 bit output port which drives a 6 bit digital to analog conversion using a weighted network of resistors. When the Mini-O programs of listing 1 and 2 store data at memory locations in the address range decoded by this port, it will be latched and appear as an equivalent voltage at the output jack. This interface is one simple way to drive a strip chart recorder monitoring the results of the Omega signal processing.

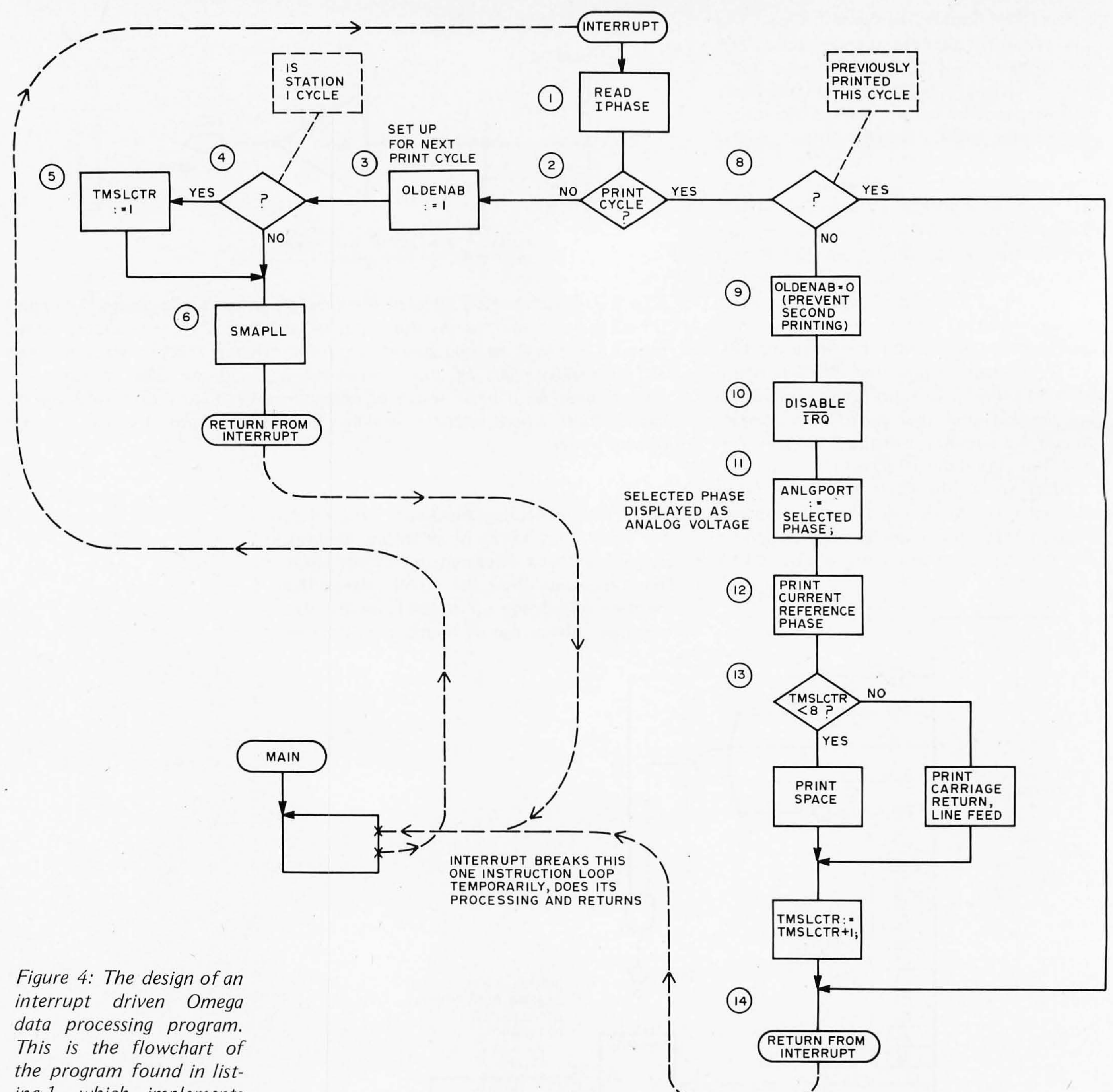


Figure 4: The design of an interrupt driven Omega data processing program. This is the flowchart of the program found in listing 1, which implements the software memory aided phase locked loop algorithm. The circled numbers are referenced in listing 1 by notations to the left of the address column.

no consequence in a small system. Furthermore if memory space becomes a valuable commodity, a thrifty alternative is to simply decode more address bits. The additional hardware required is a NOR gate, NAND gate and inverters.

The memory mapped output technique is very similar to the input scheme just described, and it is illustrated in figure 3. Again, the four most significant bits of the address bus are decoded and used to clock a latch which latches up the data on the data bus. The only differences from the input circuitry are the NANDing of the decoder output with the microprocessor "write" signal, and the data bus latch does not need to be a tristate device. The outputs of the latch are converted from digital to analog using a resistor network. This analog output can be used to drive a chart recorder or meter indicator.

Interrupt Service Routine

When the Omega data samples are entered into the microcomputer from the Mini-O interface (one sample every 25 ms), they are contaminated with noise and require filtering in order to extract a good estimate of

Table 1: SMAPLL filtered Omega Phase Data. This is a sample of the Teletype printout of the program in listing 1. Each line of the table is the result of one 10 second frame of Omega phase data compared to the internal clock. The phase differences are printed as two hexadecimal digits for each time slot, corresponding to a 6 bit number. Note that the data in the columns lettered A,B,E,F and H is quite random due to the low strength of the signals. However, the data in time slots C,D and G can be seen to be quite systematic: It increases in value as a phase difference between the internal clock and the incoming signal. By taking pairs of these well-defined station measurements and forming differences such as C-D, G-C or G-D, the result is a relatively constant value of the line of position phase difference estimate for that station pair. For example, forming the differences between station pairs G and C yields a fairly constant difference 12 average over these 24 measurements. (The mean square deviation is 3.) These observations were made at 10.2 kHz at 14:20 EDT July 19 1976.

Station = A	B	C	D	E	F	G	H
27	21	2F	29	13	2C	3E	1D
27	21	2C	2B	10	2C	3B	1E
29	21	31	2D	0F	2D	38	1F
2A	22	34	2C	0F	30	38	1D
27	22	36	2E	10	33	3A	1D
26	25	37	2E	12	33	00	1C
24	26	39	31	11	37	00	1A
25	25	3A	33	11	37	04	1C
25	24	3B	33	0F	38	09	1D
26	24	3E	35	0C	36	0C	1E
26	1F	04	37	09	37	11	20
26	21	07	37	08	38	13	20
28	23	09	3A	0A	36	15	20
2A	20	06	39	0D	3A	14	1E
2C	22	07	3C	13	39	17	1E
28	24	09	3C	13	38	19	1C
29	23	0D	3D	14	37	1B	19
2A	21	0D	3F	15	35	16	1A
2B	23	0F	01	14	36	18	19
26	23	0F	01	11	34	18	18
29	23	12	02	11	33	1C	18
28	21	14	03	17	30	20	16
28	21	14	05	16	2D	22	14
2D	1F	14	04	14	2D	21	14

the true phase. This is the function of the software version of a memory aided phase locked loop: This closed loop digital filter accepts the noisy phase measurements and provides averaging so that its output is a better estimate of the true phase.

The data format presented by the Mini-O interface is also shown in figure 1. Of the eight bits in each data byte, two are control flags and six are phase data. The most significant bit, D7, is a control signal flag which goes high to indicate when the 625 ms Omega measurement time slot window is present and data should be tracked by the software filter. The D6 flag is also controlled by the Mini-O timing hardware, and it is forced high by the Mini-O when the A time slot occurs. Bits D5 through D0 are the phase data to be tracked.

When an interrupt is received by the microprocessor, the interrupt service routine must load the new data byte, test the control bits, and perform loop filtering to the phase data bits (assuming the control bits indicated good data). Note that the microprocessor could be busily performing any main routine when it is interrupted: In our example the main program will be a simple idling loop, but it could conceivably be a more general program of use in the vehicle being navigated. A flowchart for the interrupt service routine appears in figure 4.

From this chart it can be seen that the routine must also output a filtered data word at the end of each time slot. The data output takes the form of printed numbers on the Teletype, and the memory mapped output latch with its digital to analog conversion provides a means for obtaining an analog voltage to drive a chart recorder.

Tracking Loop Software

If the D7 control bit indicates that the data is within a 625 ms time slot window, the routine must then perform loop filtering on the data via the software version of the memory aided phase locked loop. This routine is illustrated in flowchart form in figure 5. The incoming phase data is compared to the stored reference phase word. If the incoming phase is ahead of the reference phase, the difference will be positive and the loop counter must be incremented. Likewise, if the incoming phase is behind the reference phase the difference will be negative and the loop counter must be decremented. The only exception to this procedure is when the phase is near the "edge of a lane" (ie: the data points are close to 0 or 63) in which case the opposite increment or decrement must be effected.

The loop counter, which is incremented or decremented by the phase detector out-

```
      Hex
      Address  Hex Code   Label    Op         Operands        Commentary
(1)   0000     AD 00 80   START    LDA        IPHASE          Load data byte from phase measurement input;
(2)   0003     10 44               BPL        PMEAS           If high order bit zero then go print measurement;
(3)   0005     A2 80               LDX        #$80
      0007     86 E0               STX        OLDENB          OLDENB := '10000000' [set old enable flag];
(4)   0009     0A                  ASL
      000A     10 04               BPL        SMAPLL          If this is reference time slot
(5)   000C     A0 01               LDY        #1                 then set the
      000E     84 E1               STY        TMSLCTR            time slot counter to 1;
(6)   0010                * Begin software memory aided phase locked loop algorithm details
      0010     4A         SMAPLL   LSR
      0011     29 3F               AND        #$3F
      0013     85 D6               STA        PHASE           PHASE := [bits 0 to 5 of input (low order)];
      0015     A6 E1               LDX        TMSLCTR
      0017     B5 D7               LDA,X                      OLDPHREF (TMSLCTR)
      0019     4A                  LSR
      001A     4A                  LSR
(6.1) 001B     85 D7               STAA       PHIREF          PHIREF := OLDPHREF (TMSLCTR)/4;
      001D                * This adjusts for 4 increments or decrements in loop below
      001D     A5 D6               LDA        PHASE
      001F     18                  CLC
(6.2) 0020     E5 D7               SBC        PHIREF          DIFF (in accumulator) := PHASE – PHIREF;
(6.3) 0022     10 19               BPL        CKGT32P         If DIFF > 0 then go check range, positive;
      0024     A5 D7               LDA        PHIREF
      0026     18                  CLC
(6.4) 0027     E5 D6               SBC        PHASE           DIFF (in accumulator) := PHIREF – PHASE;
(6.5) 0029     A8                  TAY
      002A     A9 F6               LDA        #INC,X          Patch INC,X opcode
      002C     85 3A               STA        IDSW2              into increment or decrement switch location;
      002E     A9 D6               LDA        #DEC,X          Patch DEC,X opcode
      0030     85 37               STA        IDSW1              into increment or decrement switch location;
      0032     98         INCTST   TYA                        Begin sequence to test and increment;
      0033     C9 20               CMP        #32
(6.7) 0035     10 03               BPL        IDSW2           If | DIFF | > 32 then go to IDSW2;
(6.8) 0037     XX D7      IDSW1    DEC   or   INC             Decrement or increment OLDPHREF (TMSLCTR);
      0039     40                  RTI
(6.9) 003A     XX D7      IDSW2    INC   or   DEC             Increment or decrement OLDPHREF (TMSLCTR);
      003C     40                  RTI
(6.6) 003D     A8         CKGT32P  TAY
      003E     A9 D6               LDA        #DEC,X          Patch DEC,X opcode
      0040     85 3A               STA        IDSW2              into increment or decrement switch location;
      0042     A9 F6               LDA        #INC,X          Patch INC,X opcode
      0044     85 37               STA        IDSW1              into decrement or increment switch location;
      0046     4C 32 00            JMP        INCTST          Go to test and increment sequence;
(8)   0049     A5 E0      PMEAS    LDA        OLDENB
      004B     10 20               BPL        DONE            If previous enable flag was 0 then skip printing;
(9)   004D     A0 00               LDY        #0
      004F     84 E0               STY        OLDENB          OLDENB := [reset to prevent double printout];
(10)  0051     78                  SEI                        [mask off interrupts];
      0052     A5 DB               LDA        OLDPHREF (4)
(11)  0054     8D 00 90            STA        ANLGPORT        [analog port output := selected OLDPHREF];
(12)  0057     A5 D7               LDA        PHIREF          Print current reference phase
      0059     20 B1 72            JSR        WROB               using JOLT's TIM hexadecimal routine;
(13)  005C     A5 E1               LDA        TMSLCTR
      005E     C9 08               CMP        #8              If TMSLCTR < 8
      0060     10 06               BPL        SPCESI             then type space and increment TMSLCTR;
      0062     20 8A 72            JSR        CRLF            Print carriage return, line feed with TIM;
      0065     4C 6B 00            JMP        INCIT
      0068     20 77 73   SPCESI   JSR        SPACE           Print space using TIM routine;
      006B     E6 E1      INCIT    INC        TMSLCTR         TMSLCTR := TMSLCTR + 1 [onward to next station];
(14)  006D     40         DONE     RTI                        Wait for next interrupt from Mini-O hardware;
      006E                * main routine is an endless loop interrupted on occasion . . .
(15)  006E     4C 6E 00   MAIN     JMP        MAIN
```

Symbols: (other than labels)

```
00D6    PHASE temporary storage of 6 bit phase value
00D7    OLDPHREF origin for indexed references to old reference phase
00D7    PHIREF current 6 bit reference phase derived from table reference
00D8
 .
 .      OLDPHREF data values, indexed by TMSLCTR values 1 to 8
 .
00DF
00E0    OLDENB flag (values hexadecimal 80 or 00)
00E1    TMSLCTR current station index, 1 to 8 for stations A through H
8000    IPHASE input phase and flag data port from Mini-O peripheral
9000    ANLGPORT output analog data port for meter or chart recorder
```

Note: Location 0053 is the low order address of a phase reference value which will be displayed on the analog output channel when printing is done.

Listing 1: The Mini-O processing program. This routine implements a software memory aided phase locked loop. The bulk of the code is an interrupt response routine, starting at address 0000. Before entering the program, the IRQ interrupt vector of the JOLT or KIM-1 system must be set to point to location 0 and IRQ must be enabled. The entry point is an endless loop at location 006E.

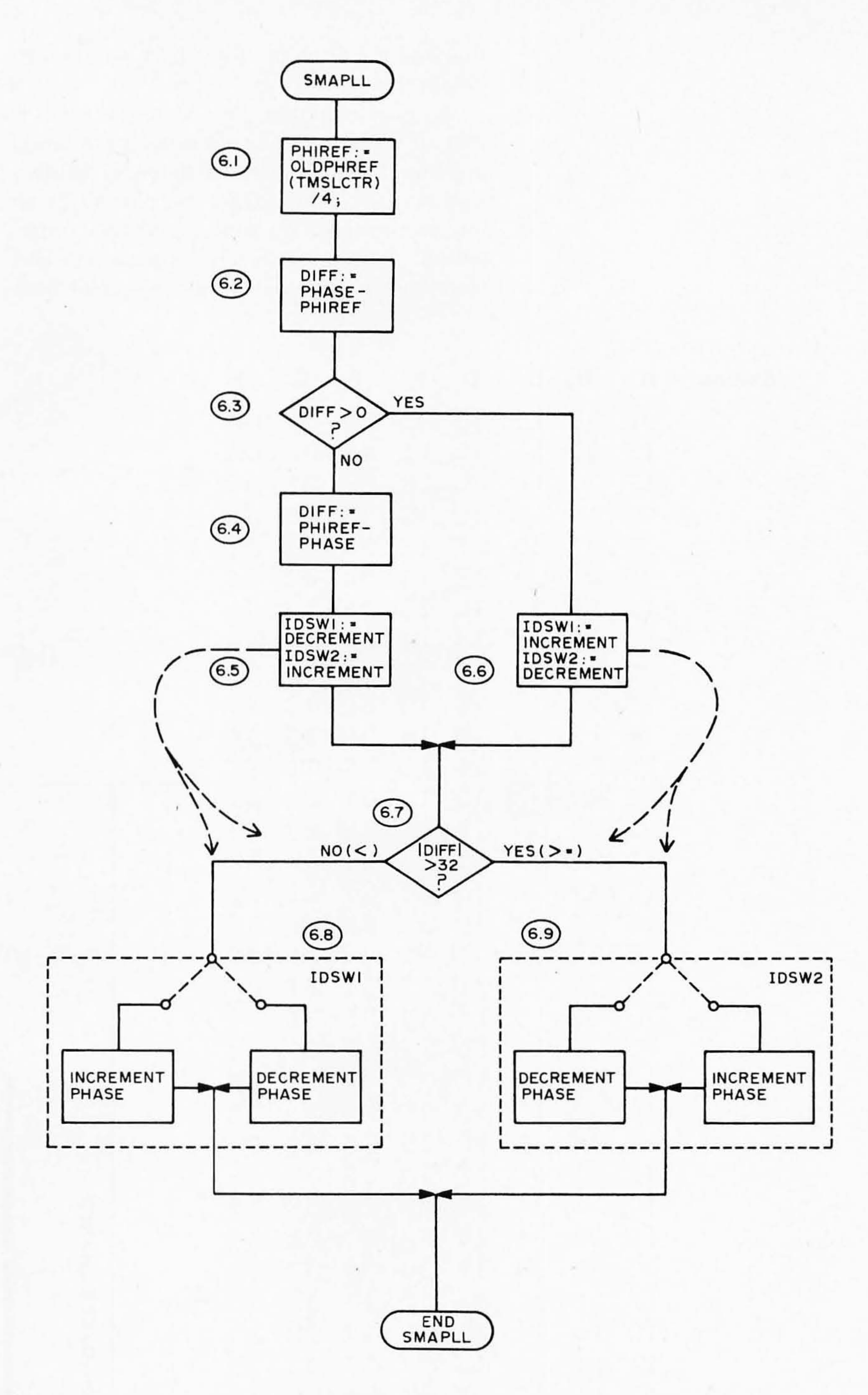


Figure 5: The detail design of the software implementation of a memory aided phase locked loop. The circled numbers here refer to listing 1. The dotted line boxes surrounding the two increment or decrement switches are an elaborate representation of the concept of a program which patches itself by changing operation codes.

put, performs the loop filtering. The length of this counter determines how much integration is provided by the SMAPLL routine, and it also sets the tracking speed of the loop. The proper choice of loop counter length is the result of an important engineering design tradeoff: If the loop counter is made too long, the noise cancelling characteristic will be enhanced but the tracking speed will be too slow to adequately track the signal. On the other hand, if the loop counter is made too short the loop tracking speed will be increased at the expense of desirable noise filtering.

The six most significant bits of the loop counter make up the filtered reference phase which is used to measure the next incoming phase word. It is also stored at the end of each time slot so that it can be used to preset the loop at the beginning of the particular time slot ten seconds later; hence, the term "memory aided" phase locked loop. During the 625 ms gap following each time slot, this reference or "locked" phase can be sent to both the Teletype printer and analog chart recorder for recording single station phase or for measuring clock offset. A typical output is shown in table 1. These reference phase words (eight total, one per time slot) can be subtracted to form the phase differences which define lines of position (LOPs) on an Omega map. Of course, this too can be output to a recording device.

Software and Output Options

The SMAPLL interrupt service routine listing for a 6502 system is given as listing 1. Omega and other VLF signals are of interest to time frequency scientists and lab experimenters because of the frequency stability of the transmitted signal. When these signals are referenced against a receiving system's local clock as in the Mini-O interface, the SMAPLL phase reference outputs can be used to determine the receiver's clock error. Similarly, any lab standard oscillator can be calibrated and adjusted to cesium atomic clock accuracy by this technique (since the Omega signals are locked to a cesium reference at each transmitter). Table 2 illustrates this method of clock offset determina-

tion and the program of listing 1 was used to obtain this output.

Another common use of Omega signals is that of position fixing. In order to obtain a position fix, the phase differences of two station pairs (two LOPs) are plotted as an intersecting grid on a map and their intersection defines the receiver's position. This technique is suitable for use by small boat operators or backpackers. In order to plot a line of position, the segment of the program in listing 2 must be substituted into listing 1 beginning at location 0052. The LOP output obtained with this program and plotted on a Rustrak chart recorder appears in figure 8.

In order to compare the operation of the software version of a memory aided phase locked loop and a conventional hardware loop system, the routines were run in a JOLT microcomputer side by side with an all hardware Omega sensor and processor. The lines of position outputs of both systems were plotted on a dual trace Rustrak chart recorder in our lab. This chart appears in figure 6 with the software system output *on* the bottom. As can be seen from this chart, the software system tracks the Omega signal in very similar manner to the hardware system.

Station = A	B	C	D	E	F	G	H
20	00	19	25	23	29	39	0A
1F	00	1A	29	1E	2E	3A	0A
1F	01	1D	2B	1D	30	3F	0B
21	04	23	2E	18	33	05	0B
21	09	29	2F	13	34	0A	07
1F	0B	2B	33	13	34	0C	06
1E	11	32	35	10	35	12	08
20	16	36	38	11	34	14	05
23	19	39	39	0E	37	16	01
23	1B	38	3C	13	3D	18	3D
1F	1F	39	3D	16	3D	1E	3F
20	25	3A	01	18	3F	1E	3F
22	28	**3F**	03	1B	01	21	02
21	2D	04	05	1E	04	25	03
1D	2D	09	07	20	01	24	02
1E	32	0B	0A	23	3D	25	01
1E	37	0B	0D	22	3D	2A	3C
1C	3C	0D	0F	25	3A	2F	33
18	39	0F	11	27	3C	31	33
1B	3D	10	15	27	35	35	30
18	3C	14	16	26	32	36	30
1B	3F	16	18	2B	2F	39	2F
1C	03	1C	1B	2F	2E	3B	32
1E	3F	1E	1D	35	2C	3F	34
1E	03	1F	1F	3A	29	02	33
1D	05	21	23	3D	2B	00	32
1E	0A	26	24	3B	2D	04	34
20	0F	29	28	00	32	09	34
1F	0F	2D	2B	02	32	08	34
1C	14	2E	2D	01	37	08	34
18	14	2B	2B	3F	38	08	35
17	14	2D	30	02	3A	0A	36
14	12	2C	34	02	37	0E	35
17	12	2E	38	03	37	0D	34
1B	10	31	3A	02	3A	11	34
18	12	33	3E	05	35	12	33
1A	16	32	3F	05	36	11	34
19	13	36	04	03	34	13	33
18	14	37	05	0B	36	14	32
18	12	3A	03	0B	34	17	30
19	12	3D	0B	10	2F	1B	2F
1C	0F	**3F**	0F	12	2D	1C	2B
21	10	04	0F	12	2D	1A	2F
21	10	08	15	15	2B	1F	32
24	0B	0C	16	16	29	1C	30
26	08	0F	1C	16	27	1D	2F

ONE CYCLE CHANGE

Future Software

Line of position outputs and phase versus clock information can serve the boat operator and time frequency experimenter respectively, but airborne navigators need position data in a more suitable form (such as cross-track error, miles-to-go, etc). Development of the necessary navigation software to enable the accomplishment of these tasks with an inexpensive, commerically available

$$\text{Clock Offset} = \frac{\text{One Omega Cycle time}}{\text{Time to Change One Cycle}}$$

$$= \frac{1/10200 \text{ seconds}}{290 \text{ seconds}}$$

$$= \frac{09.80 \times 10^{-5}}{2.90 \times 10^{2}}$$

$$= 3.38 \times 10^{-7}$$

Table 2: Local clock offset determination. By measuring the amount of time needed for phase to change one complete cycle, relative to the local clock, it is possible to measure the difference between your local clock reference and the Cesium atomic clocks used to calibrate the Omega system at its world wide network of stations. The calculation is shown here, along with data taken from 10.2 kHz stations at 10:30 EDT July 20 1976. This is one way to achieve the ultimate in personal measurement of time using your computer and an Omega receiver.

Hex Address	Hex Code	Label	Op	Operand	Commentary
0052	E0 04	PATCH	CPX	#4	Is old TMSLCTR = 4 [selected station D index] ?
0054	F0 18		BEQ	DIFCALC	If so then go to DIFCALC [calculate difference] ;
0056	EA		NOP		Fill space;
0057	......	ENDPTCH			Return to previous code;

006E	A5 DA	DIFCALC	LDA	OLDPHREF (3)	
0070	4A		LSR		
0071	4A		LSR		
0072	85 D3		STA	TPHAZ	TPHAZ := OLDPHREF(3)/4 [get C phase] ;
0074	A5 D7		LDA	PHIREF	
0076	18		CLC		
0077	E5 D3		SBC	TPHAZ	DIFF := PHIREF – TPHAZ;
0079	30 08		BMI	NEGAT	If negative then go to NEGAT;
007B	0A	OUTDIFF	ASL		
007C	0A		ASL		Shift left to correspond to DAC output bits;
007D	8D 00 90		STA	ANLGPORT	[analog output port := difference phase] ;
0080	4C 57 00		JMP	ENDPTC	Jump back in line;
0083	69 40	NEGAT	ADC	#$40	[convert to positive difference] ;
0085	4C 7B 00		JMP	OUTDIFF	Go to OUTDIFF [and display value] ;
0088	4C 88 00	MAIN	JMP	MAIN	

Additional symbols (other than labels):

00D3	TPHAZ	temporary byte used in phase difference calculation

Listing 2: An alternate version of the software memory aided phase lock loop for use in plotting a phase difference value versus time. This patch to listing 1 was used in preparing the chart of figure 6. It calculates the phase difference, between stations selected by the choice of addresses D8 to DF at address 006F, and the choice of a station index 1 to 8 at location 0053. (The ranges in each case correspond to station selections A through H.) The setup shown here picks the C-D station pair. In preparing figure 6, the constants were set up for the G-C station pair. With this patch, the main entry point is now at location 0088.

microprocessor is a continuing effort at Ohio University. The microcomputer has made a low cost Omega Navigation System a very real possibility and BYTE readers are advised to stand by for navigation software as it is perfected.■

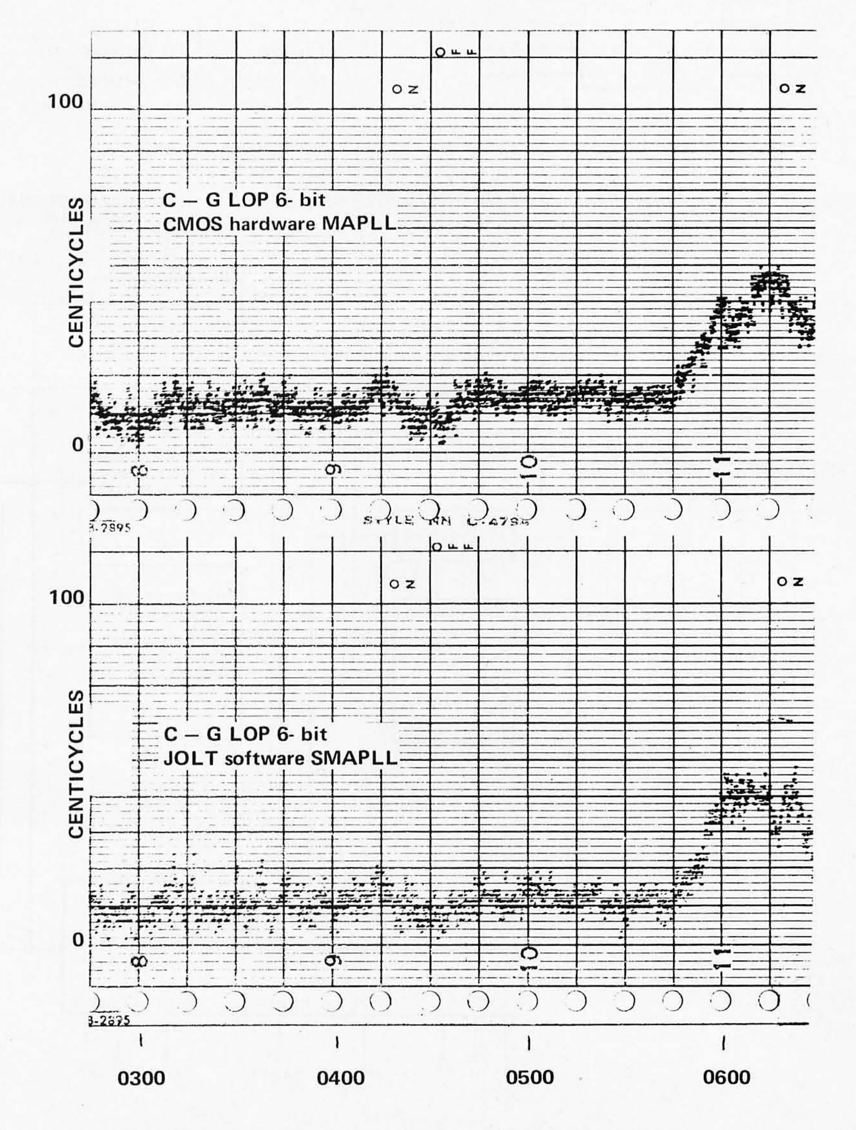


Figure 6: Comparison of the Mini-O software memory aided phase locked loop (listing 1) with a comparable hardware version in older (more expensive) equipment. The two sets of equipment were run simultaneously beginning at 3 PM EDT on July 16 1976 with the 2 channel chart recorder running at one inch per hour. Timing calibration is shown at the bottom of the chart.

Microprocessor Update: 8008

Robert Baker
15 Windsor Dr
Atco NJ 08004

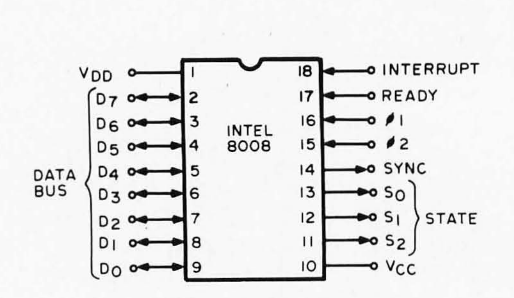


Figure 1: The pinouts of the Intel 8008 processor.

For the newcomer to the microcomputer world, the 8008 was one of the first 8 bit microprocessors to be developed. Figure 1 shows the pin assignment of the 18 pin dual-in-line package while figure 2 shows a block diagram of the central processor unit (CPU). Typical prices for the 8008 CPU are currently under $20.

The 8008 is a single chip, MOS, 8 bit CPU that communicates over an 8 bit data and

Figure 2: Internal organization of the 8008 processor.

MNEMONIC	MINIMUM STATES REQUIRED	INSTRUCTION CODE D_7D_6	$D_5D_4D_3$	$D_2D_1D_0$	DESCRIPTION OF OPERATION
(1) Lr_1r_2	(5)	1 1	D D D	S S S	Load index register r_1 with the content of index register r_2.
(2) LrM	(8)	1 1	D D D	1 1 1	Load index register r with the content of memory register M.
LMr	(7)	1 1	1 1 1	S S S	Load memory register M with the content of index register r.
(3) LrI	(8)	0 0 B B	D D D B B B	1 1 0 B B B	Load index register r with data B . . . B.
LMI	(9)	0 0 B B	1 1 1 B B B	1 1 0 B B B	Load memory register M with data B . . . B.
INr	(5)	0 0	D D D	0 0 0	Increment the content of index register r (r ≠ A).
DCr	(5)	0 0	D D D	0 0 1	Decrement the content of index register r (r ≠ A).

Accumulator Group Instructions

The result of the ALU instructions affect all of the flag flip-flops. The rotate instructions affect only the carry flip-flop.

MNEMONIC	MINIMUM STATES REQUIRED	INSTRUCTION CODE D_7D_6	$D_5D_4D_3$	$D_2D_1D_0$	DESCRIPTION OF OPERATION
ADr	(5)	1 0	0 0 0	S S S	Add the content of index register r, memory register M, or data B . . . B to the accumulator. An overflow (carry) sets the carry flip-flop.
ADM	(8)	1 0	0 0 0	1 1 1	
ADI	(8)	0 0 B B	0 0 0 B B B	1 0 0 B B B	
ACr	(5)	1 0	0 0 1	S S S	Add the content of index register r, memory register M, or data B . . . B to the accumulator with carry. An overflow (carry) sets the carry flip-flop.
ACM	(8)	1 0	0 0 1	1 1 1	
ACI	(8)	0 0 B B	0 0 1 B B B	1 0 0 B B B	
SUr	(5)	1 0	0 1 0	S S S	Subtract the content of index register r, memory register M, or data B . . . B from the accumulator. An underflow (borrow) sets the carry flip-flop.
SUM	(8)	1 0	0 1 0	1 1 1	
SUI	(8)	0 0 B B	0 1 0 B B B	1 0 0 B B B	
SBr	(5)	1 0	0 1 1	S S S	Subtract the content of index register r, memory register M, or data B . . . B from the accumulator with borrow. An underflow (borrow) sets the carry flip-flop.
SBM	(8)	1 0	0 1 1	1 1 1	
SBI	(8)	0 0 B B	0 1 1 B B B	1 0 0 B B B	
MNEMONIC	**MINIMUM STATES REQUIRED**	**INSTRUCTION CODE D_7D_6**	**$D_5D_4D_3$**	**$D_2D_1D_0$**	**DESCRIPTION OF OPERATION**
NDr	(5)	1 0	1 0 0	S S S	Compute the logical AND of the content of index register r, memory register M, or data B . . . B with the accumulator.
NDM	(8)	1 0	1 0 0	1 1 1	
NDI	(8)	0 0 B B	1 0 0 B B B	1 0 0 B B B	
XRr	(5)	1 0	1 0 1	S S S	Compute the EXCLUSIVE OR of the content of index register r, memory register M, or data B . . . B with the accumulator.
XRM	(8)	1 0	1 0 1	1 1 1	
XRI	(8)	0 0 B B	1 0 1 B B B	1 0 0 B B B	
ORr	(5)	1 0	1 1 0	S S S	Compute the INCLUSIVE OR of the content of index register r, memory register m, or data B . . . B with the accumulator.
ORM	(8)	1 0	1 1 0	1 1 1	
ORI	(8)	0 0 B B	1 1 0 B B B	1 0 0 B B B	
CPr	(5)	1 0	1 1 1	S S S	Compare the content of index register r, memory register M, or data B . . . B with the accumulator. The content of the accumulator is unchanged.
CPM	(8)	1 0	1 1 1	1 1 1	
CPI	(8)	0 0 B B	1 1 1 B B B	1 0 0 B B B	
RLC	(5)	0 0	0 0 0	0 1 0	Rotate the content of the accumulator left.
RRC	(5)	0 0	0 0 1	0 1 0	Rotate the content of the accumulator right.
RAL	(5)	0 0	0 1 0	0 1 0	Rotate the content of the accumulator left through the carry.
RAR	(5)	0 0	0 1 1	0 1 0	Rotate the content of the accumulator right through the carry.

Program Counter and Stack Control Instructions

(4) JMP	(11)	0 1 B_2 B_2 X X	X X X B_2 B_2 B_2 B_3 B_3 B_3	1 0 0 B_2 B_2 B_2 B_3 B_3 B_3	Unconditionally jump to memory address $B_3 \ldots B_3B_2 \ldots B_2$.
(5) JFc	(9 or 11)	0 1 B_2 B_2 X X	0 C_4 C_3 B_2 B_2 B_2 B_3 B_3 B_3	0 0 0 B_2 B_2 B_2 B_3 B_3 B_3	Jump to memory address $B_3 \ldots B_3B_2 \ldots B_2$ if the condition flip-flop c is false. Otherwise, execute the next instruction in sequence.
JTc	(9 or 11)	0 1 B_2 B_2 X X	1 C_4 C_3 B_2 B_2 B_2 B_3 B_3 B_3	0 0 0 B_2 B_2 B_2 B_3 B_3 B_3	Jump to memory address $B_3 \ldots B_3B_2 \ldots B_2$ if the condition flip-flop c is true. Otherwise, execute the next instruction in sequence.
CAL	(11)	0 1 B_2 B_2 X X	X X X B_2 B_2 B_2 B_3 B_3 B_3	1 1 0 B_2 B_2 B_2 B_3 B_3 B_3	Unconditionally call the subroutine at memory address $B_3 \ldots B_3B_2 \ldots B_2$. Save the current address (up one level in the stack).
CFc	(9 or 11)	0 1 B_2 B_2 X X	0 C_4 C_3 B_2 B_2 B_2 B_3 B_3 B_3	0 1 0 B_2 B_2 B_2 B_3 B_3 B_3	Call the subroutine at memory address $B_3 \ldots B_3B_2 \ldots B_2$ if the condition flip-flop c is false, and save the current address (up one level in the stack.) Otherwise, execute the next instruction in sequence.
CTc	(9 or 11)	0 1 B_2 B_2 X X	1 C_4 C_3 B_2 B_2 B_2 B_3 B_3 B_3	0 1 0 B_2 B_2 B_2 B_3 B_3 B_3	Call the subroutine at memory address $B_3 \ldots B_3B_2 \ldots B_2$ if the condition flip-flop c is true, and save the current address (up one level in the stack). Otherwise, execute the next instruction in sequence.
RET	(5)	0 0	X X X	1 1 1	Unconditionally return (down one level in the stack).
RFc	(3 or 5)	0 0	0 C_4 C_3	0 1 1	Return (down one level in the stack) if the condition flip-flop c is false. Otherwise, execute the next instruction in sequence.
RTc	(3 or 5)	0 0	1 C_4 C_3	0 1 1	Return (down one level in the stack) if the condition flip-flop c is true. Otherwise, execute the next instruction in sequence.
RST	(5)	0 0	A A A	1 0 1	Call the subroutine at memory address AAA000 (up one level in the stack).

Input/Output Instructions

INP	(8)	0 1	0 0 M	M M 1	Read the content of the selected input port (MMM) into the accumulator.
OUT	(6)	0 1	R R M	M M 1	Write the content of the accumulator into the selected output port (RRMMM, RR ≠ 00).

Machine Instruction

HLT	(4)	0 0	0 0 0	0 0 X	Enter the STOPPED state and remain there until interrupted.
HLT	(4)	1 1	1 1 1	1 1 1	Enter the STOPPED state and remain there until interrupted.

NOTES:

(1) SSS = Source Index Register, DDD = Destination Index Register — These registers, r_i, are designated A(accumulator—000), B(001), C(010), D(011), E(100), H(101), L(110).

(2) Memory registers are addressed by the contents of registers H & L.

(3) Additional bytes of instruction are designated by BBBBBBBB.

(4) X = "Don't Care".

(5) Flag flip-flops are defined by C_4C_3: carry (00-overflow or underflow), zero (01-result is zero), sign (10-MSB of result is "1"), parity (11-parity is even).

Table 1: The instruction set of the Intel 8008, reproduced from Intel's November 1973 8008 central processor unit User's Manual, *pages 8 and 9.*

Continued on page 160

The Best of
creative computing
Volume 1 Edited by David H. Ahl

The diversity in *The Best of Creative Computing — Volume 1* can only be described as staggering. The book contains 328 pages of articles and fiction about computers, games that you can play with computers and calculators, hilarious cartoons, vivid graphics and comprehensive book reviews.

Authors range from Isaac Asimov to Sen. John Tunney of California; from Marian Goldeen, an eighth-grader in Palo Alto to Erik McWilliams of the National Science Foundation; and from Dr. Sema Marks of CUNY to Peter Payack, a small press poet. In all, over 170 authors are represented in over 200 individual articles, learning activities, games, reviews and stories.

This 328-page book has 108 pages of articles on computers in education, CAI, programming, and the computer impact on society; 10 pages of fiction and poetry including a fascinating story by Isaac Asimov about all the computers on earth linking up after a nuclear war to support the few remaining survivors; 15 pages of "Foolishness" including a cute cartoon piece - called "Why We're Losing Our War Against Computers"; 26 pages on "People, Places, and Things" including the popular feature "The Compleat Computer Catalogue" which gives capsule reviews and lists sources for all kinds of computer-related goodies; 79 pages of learning activities, problems and puzzles; 29 pages continuing 18 computer games including a fantastic extended version of the single most popular computer game — Super Star Trek; and 32 pages of in-depth book and game reviews including Steve Gray's definitive review of 34 books on the Basic language.

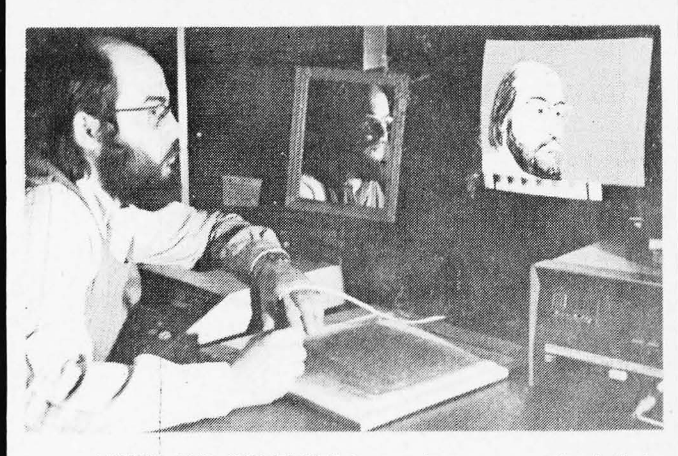
The Best of Creative Computing - Volume 1 is available by mail for $8.95 plus 75¢ postage from Creative Computing Press, Dept. B-21, P.O. Box 789-M, Morristown, N.J. 07960.



THE BEST OF BYTE — VOL. 1

The Best of Byte - Volume 1 is a 384-page blockbuster of a book which contains the majority of material from the first 12 issues of *Byte* magazine. 146 pages are devoted to "Hardware" and are cram full of how-to articles on everything from TV displays to joysticks to cassette interfaces. The section on computer kits describes building 7 major kits. But hardware without software might as well be a boat anchor, so there are 125 pages of "Software and Applications" ranging from on-line debuggers to games to a complete small business accounting system. A section on "Theory" examines the how and why behind the circuits and programs, and a final section "Opinion" looks at where this explosive new hobby is heading.

The Best of Byte - Volume 1 is edited by Carl Helmers and David Ahl and published by Creative Computing Press. Price in the US is $11.95 plus $1.00 shipping and handling ($12.95 total); foreign orders add $1.00 ($13.95 total). Orders from individuals must be prepaid. Creative Computing Press, Dept. B-21, P.O. Box 789-M, Morristown, NJ 07960. Allow 8 weeks for delivery.



ARTIST AND COMPUTER is a unique new art book that covers a multitude of computer uses and the very latest techniques. In its pages, 35 artists who work with computers explain how the computer can be programmed either to actualize the artist's concept (such as the visualization of fabric before it is woven) or to produce finished pieces. Illustrated with more than 160 examples of computer art, 9 of them in full color, ARTIST AND COMPUTER will fascinate and inspire anyone who is interested in art or computer technology. Size 8½" x 11".

ARTIST AND COMPUTER

Edited by RUTH LEAVITT

Paper $4.95, cloth $10; now at selected bookstores, or send payment plus 75¢ handling to Creative Computing, P.O. Box 789-M, Morristown, N.J. 07960. N.J. residents add 5% sales tax.



101 BASIC Computer Games is the most popular book of computer games in the world. Every program in the book has been thoroughly tested and appears with a complete listing, sample run, and descriptive write-up. All you need add is a BASIC-speaking computer and you're set to go.

101 BASIC Computer Games. Edited by David H. Ahl. 248 pages. 8½x11 paperbound. $7.50 plus 75¢ postage and handling ($8.25 total) from Creative Computing, P.O. Box 789-M, Morristown, NJ 07960.



Guess who just bought our new Microfloppy™ for $1095 including software?

Plug-compatible for $1095

iCOM's brand new Microfloppy™ is specifically designed for the personal computer user. It's smaller, using the new 5¼-inch diskette, and is priced at only $1095 including disk drive, power supply, cabinet, controller/interface card, all cables and connectors, plus, on diskette, iCOM's new FDOS-M software, including assembler, editor, and file manager. And for a limited time,* you also get an 8K Disk BASIC package at no extra cost.

Best of all, iCOM's Microfloppy™ is fully assembled and tested. It's 100% compatible with Altair, IMSAI, Poly 88 and other microcomputers using the Altair bus format.

*Offer good through April 15, 1977



Much faster than a cassette

This compact, economical Floppy Disk System is 10 to 15 times faster than a cassette. Programs can be loaded in seconds, a source program of several thousand characters can be corrected and a new file stored on diskette in less than a minute. Any of up to 175 files can be reached in 1.5 seconds.

iCOM — the leader

We've been building Floppy Disk Systems for microcomputers for nearly three years. Thousands of our systems are operating perfectly in the field. Many major computer manufacturers have incorporated our Floppy Disks into their systems. We deliver, and we'll be around to give you service whenever you may need it. You see, we're a division of Pertec Computer Corp., one of the largest and most respected manufacturers of peripherals, microsystems, data entry products, and data processing systems.

So you don't need to risk getting burned by a fly-by-night company. Order an iCOM® Microfloppy™ and be sure of getting the best.

See iCOM's Microfloppy™ now at your local computer store. Or contact us for the name of your dealer.

icom® MICROPERIPHERALS®

a division of Pertec Computer Corporation

6741 Variel Ave., Canoga Park, CA 91303 U.S.A.
(213) 348-1391 TWX 910-494-2788

Continued from page 32

bibliographic information about a manuscript and the second containing the present working text buffer. Or consider the problem of remembering one section of code while editing a program electronically – then altering a second portion at a different place. The idea of multiple text editing windows via separate displays is a very powerful concept which is well worth exploring.

As to split screen displays where lines are intended to span several screens, there is a fundamental problem: The gap between the display tubes interferes with one's scanning of the text. . . CH

YET ANOTHER DEFINITION OF THE WORD BYTE . . . AND BAR CODE COMMENTS

In reference to Thomas P Bishop's letter *[page 77]*, published in November 1976 BYTE: One of my pet quirks is labeling things. Like so many authors of today's computer texts, I put a label on everything. I would, however, like to clarify the meaning of the term byte; so, in that connection, I have a bit (eight to be exact) of information. According to Donald E Knuth, the father of many of today's algorithms, a byte is not related to IBM (it is about the only thing in the computing world which isn't) but rather it is a universal concept.

> *The basic unit of information is a byte. Each byte contains an unspecified amount of information, but it must be capable of holding at least 64 distinct values. That is, we know that any number between 0 and 63, inclusive, can be contained in one byte. Furthermore, each byte contains at most 100 distinct values. On a binary computer a byte must therefore be composed of six bits; on a decimal computer we have two digits per byte.*

This is an item of information which is invaluable, or should be invaluable to Mr Bishop, since I have just relieved his "plight." Furthermore, Mr Helmers should also be pleased, since now an explanation has been found for the name of the magazine he created.

I am extremely pleased with the quality and quantity of BYTE, and I would like to say "keep assembling, compiling and interpreting these fascinating bits, bytes and words of data," ie: good work, and keep it up!

One last question: Do you know of any company presently manufacturing a hand-held scanner capable of reading the future bar codes which will be published by BYTE?

R H Beinert
223 Du Rose Ter
Madison WI 53705

See the letter by W Buchholz [February 1977 BYTE, page 144] *and note*

also that your quote from Knuth makes no reference to the origin of the term but simply generalizes the definition to mean any mouthful of data with greater than 6 bits in a binary representation.

The scanners have not yet been produced in commercial form, but we have been contacted by several firms intending to do so. The first firm which achieves the $20 scanner (or at least the same order of magnitude price) capable of transforming bars as small as 0.01 inch (0.25 cm) in width into TTL level signals will most likely get very good publicity as the prize for winning the race. We intend to support the concept, with good software starting with Jack Emmerichs' assembler published in this issue of BYTE and interpreter and monitor software as well as miscellaneous applications, fun and games. As a rule of thumb, PAPERBYTES™ products will be distributed in both source listing *and* object *bar code forms unless specific arrangements to the contrary are made by program authors.*

CODE DECODED . . . COMMENTS ON MORSER

I hope I'm not too late to get my vote in for Grappel and Hemenway's October 1976 article, page 30. I believe it was the best I've ever seen in BYTE and I look forward to every issue. I phoned Mr Grappel to thank him personally, but I'm sure he'd just as soon have the $100 award.

I enjoyed this article because it described a simple starting point for what can be an almost unsolvable problem. The biggest obstacle in any accomplishment is usually getting started. I suspected there were others who thought about such an idea but let the possible complications keep them from writing that first line of code. I've made some changes to the program but I really appreciated being pointed in the right direction. Thanks to all.

In my conversation with Mr Grappel, he suggested a replacement for the all important audio input processing circuit designed around the NE567PLO tone decoder. I had very little luck with these tone decoder chips. Low noise immunity seems to be their problem. Mr Grappel said the one in *QST*, November 75, page 15, looked much more promising. I built this unit up and had considerably more success. The unit in the *QST* article by Thomas Riley uses a bandpass filter which I have *not* yet incorporated and still have obtained good results. The signal still has to be pretty clean but the Schmitt trigger or the *QST* circuit has much more immunity than the tone decoder. When the filter is added I expect to see more improvement. I toyed with the idea of getting my computer a code proficiency certificate from W1AW when it copied a minute solid at 20 words per minute. That was much better than normal. The receiver noise was quiet and the W1AW signal was quite strong.

Incidentally, the only ICs used in the improved audio processor are a pair of NE555s (U33, U32) and an LED. Output is in the right sense at pin 3 of U33. Missing pin number 7 on U32 must be tied to pin 6 or it will never work.

Please keep up the good work of BYTE. I hope these next three years are half as interesting as the first.

Bill Bennett K3TNM
1908 Foxhound Ct
Severn MD 21144

COMPUTER COURSES WANTED

I completed three years of active duty in the US Army approximately 2-1/2 years ago. During this 2-1/2 years, I have been keeping an eye out for an educational program in which I could apply my VA benefits.

Since I am *very* interested in owning a personal computer with CRT, disk drives, etc, etc, and learning about the principles of the processor and all of its supporting and peripheral equipment, I would like to find a course to which I could apply my VA benefits along these lines. Do you, one of BYTE's readers or advertisers, know of such a course or courses?

If a company could develop such a course around its computer and peripheral kits, it could expand its market to those who are very interested in personal size computers, but do not have the knowledge and/or funds to own one at the present.

I would appreciate any help from anyone who could get me in contact with such a course.

John A Whitney
3644 Parkway Pl
Shreveport LA 71104

1802 USERS CLUB WANTED

I just received the November BYTE. Wow! Some good coverage of video graphics. Keep the "display" articles coming. (There may be a few that felt there was too much emphasis on graphics; but on the other hand, I'd wager the majority could stand a lot more.)

I'm one of those "homebrewers" and am in the midst of getting my computer act together. It will be housed in an attache case, use an RCA 1802 processor, 4 K (minimum) or prorammable memory, a keyboard, TV and cassette recorder interfaces, of course. Everything possible will be CMOS logic to minimize power supply requirements, with either 110 VAC or battery power.

I was *someday* going to get into personal computing; then the August

1976 BYTE bit me with Suding's television display device. I studied it thoroughly to learn all the required signals to generate television video. Then I converted the design to my requirements: CMOS wherever possible, a single supply character generator using +5 V only, 21L02s for memory, etc. My result now takes 26 chips instead of 32, cost $45 in parts including the crystal and wire wrap board (found in a surplus shop) and draws only 5 V at 250 mA.

Now, a question: Is there anyone out there using the RCA 1802 processor chip or is it too new for the hobbyist? Could you print a short request for responses from any 1802 users? I think this may be of interest to many others besides myself.

Harley Shanko
Apartment 209
15025 Vanowen St
Van Nuys CA 91405

ON USING BURIED GOLD

I don't own an Altair 8800 or whatever (yet), but I do have an SR-52 of my very own and I was pleased to see "The Buried Gold in the SR-52," by Clif Penn, page 30, and "Desk Top Wonders," page 92, in the December BYTE. When I heard about the hidden goodies in the SR-52 three months ago, I wrote to Texas Instruments and received a prompt reply with a printed sheet giving much the same dope on the registers as in Clif Penn's story. Although the first level of the stack is addressable as R60, it actually cannot be used if you are doing any arithmetic with dyadic operators (plus, minus, times, divided by, y to the x power or x root of y) because the number on the left hand side of the operator is written into the first level of the stack (register 60). So in most cases only R61 to R69 are available. This I found from personal tinkering.

Would it be possible to publish the address of the SR-52 Users Club and *52 Notes*? For that matter, I think it would be a good feature to publish a list of users' groups for the various microcomputer systems and programmable calculators.

The big advantage of programmable calculators is that they are portable. But are they inexpensive in terms of computing power per dollar? Polymorphic Systems offers a ready to run system with an 8080 based processor, 8 K memory, video interface, cassette interface, keyboard, television monitor, cassette recorder, BASIC and an assembler for $1750, or about six times the list of an SR-52. The Poly 88 is faster by several times and has several times the programming capacity, not to mention the ability to handle nonnumeric data, and can be expanded. I would say you get more than six times the computing power. There are certainly less expensive systems in ready to run or kit form available from other manufacturers.

Texas Instruments, I've heard, has a

super SR-52 desk top machine called the SR-61. Could you run a product description on it in a forthcoming issue of BYTE?

William R Hamblen
946 Evans Rd
Nashville TN 37204

SR-52 Users Club is at 9459 Taylorville Rd, Dayton OH 45424, (513) 233-3698, and publishes 52 Users Notes *monthly.*

A CALCULATOR USER'S COMMENTS

As a member of the SR-52 Users Club, I was pleased to read "The Buried Gold in the SR-52" by Clif Penn, page 30; also "Desk Top Wonders – Shooting Stars," page 92; and letter "New Product: Snob Detector" by Craig Pearce, page 53, in the December BYTE #16. I hope you have more on the SR-52 as well as other programmable calculators such as the SR-56 and the HP-25.

I think that the idea of machine readable printed software on page 12, etc, of the same issue of BYTE is a great idea and should be a great help in the production of low cost software. It should also be a good idea for software for programmable calculators. Those programmable calculators that are not keyboard programmable only are programmed by magnetic cards except for some of the "super" calculators such as advanced members of the HP-9800 series. Magnetic cards are small and easy to lose. They are expensive as compared to your new form of printed software. They cannot be duplicated as easily as printed software. They can lose a bit or two over a long period of time, but printed software should last longer and it should be more obvious when something is wrong with a printed program, also cheaper to replace. A low cost optical reader is needed to make the idea of machine readable printed software economical.

I have recently been reading *Computers at Large* by Charles J Sippl and Robert Bullen (Bobbs-Merrill, NY 1976). Starting at the beginning of chapter four, "The Robots are Coming – and Giants Too," there is a description of a robot that can learn routines by having the operator run the machine through a routine manually, once, and the machine has "learned" the routine and can repeat it as it was done the first time. In doing this the robot has "written" a program and developed an algorithm for the routine. It has done this a lot quicker than a human being could do it and a lot cheaper, too. It seems to me that this idea could be expanded for mathematical modeling and simulation, beyond its original use for industrial routines.

David W Johnston
POB 3781
Washington DC 20007

ITTY BITTY?

I'm writing you in regard to your September issue editorial, "Come One, Come All!", page 6.

In your article you mentioned Ted Nelson and his "itty bitty machine corporation." I would like to know how I can communicate with Mr Nelson. Could you please send me his mailing address.

Your response to my request would be greatly appreciated, as I don't know when or if I'll get your magazine again.

Ira Ford
19826 48th Av W
#L-9
Lynnwood WA 98036

See the advertisements for itty bitty and Data Domain beginning in January BYTE on page 63.

WHERE'S A TWO ADDRESS MACHINE?

Will there ever (in the next few years) be a two address microprocessor on the market? I have been using (in school and in work) several of the NCR Century series computers, which by the way are two address machines, for the past five years. I have also used single address machines (ie: Altair 8800, IBM 1130 and CHI 2130), but generally I have found them a pain to work with after the NCRs. I just can't see the point of having a set of 78 hardware commands to perform memory to register, register to register and register to memory functions when a set of 39 memory to memory commands are sufficient. OK, I'll admit it is unfair to compare a medium scale computer with a mini or microprocessor, but, the NCR line was developed for the *1968* sales market without the LSI techniques now in use. There is place for both the single and double address machines, but I feel the ease in programming of a two address processor makes it far superior to a single address processor. I am not putting the single address machines down, but I feel they are for the same people who would prefer to use an RPN calculator when there is an AOS calculator available. Let the machine work for you! Anyway, I haven't bought a micro yet because I am waiting; I have a feeling it will be a long wait, for a two address machine. CAN ANYONE HELP??

Jon Kirkpatrick
Fullerton CA

Want a two address machine? Take a look at a microprogram oriented architecture using contemporary bipolar microcontrollers such as Signetics 8X300 or AMD 2900 series parts. Then do your



MICROCOMPUTING FOR HOME AND THE SMALL BUSINESSMAN

The professional publication bringing microcomputing technology to the hobbyist, small businessman, educator, engineer and student. Every issue edited to bring technology and people together in the simplest manner.

INTERFACE AGE is packed with

- **HARDWARE ARTICLES** — Product profiles, comparisons, applications, modifications and construction projects.
- **SOFTWARE ARTICLES** — Microcomputer development software, short software routines, application software, off-line software storage formats, software communication standards and access to the microcomputer software depository for all **INTERFACE AGE** readers.
- **TUTORIALS** — Fundamentals of micro processors, basics for microcomputing, professional to technical transition information, understanding software, elementary math for computing.
- **NEW PRODUCTS** — Manufacturer profiles and latest product releases.
- **USER COMMUNICATION** — ***UPDATE*** — devoted to club and organization announcements and activities. — ***LETTERS TO THE EDITOR*** — Reader forum for expression of opinions and feedback on articles and features.
- **MICRO-MARKET/FIFO FLEA MARKET** — Low cost/no cost advertising for the new marketeer/garage sale enthusiast.

If you need to know how to get started in microcomputing or need the valuable software once your system is completed then **INTERFACE AGE** is a must for you.

Don't Delay — Subscribe Today!

INTERFACE AGE MAGAZINE

P.O. Box 1234
Cerritos, CA. 90701

12 Monthly Issues:
$10 U.S., $12 Can./Mex.,
$18 International

☐ CHECK ☐ MONEY ORDER

Name ____________________

Address ____________________

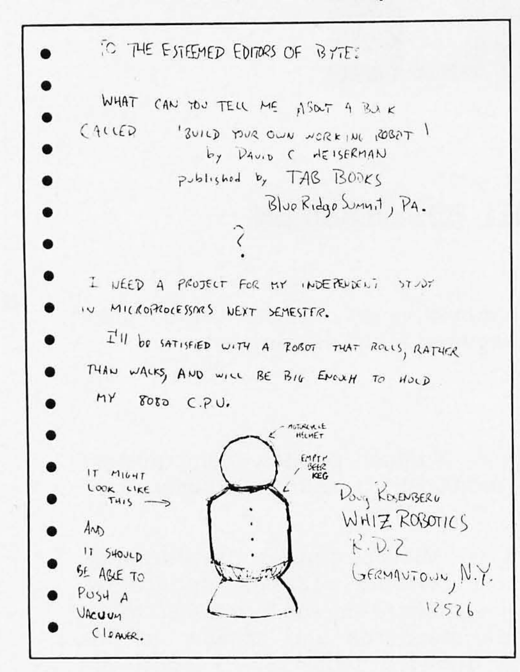
City ____________________ State ______ Zip ______

Country ____________________

MAKE CHECK PAYABLE TO: **INTERFACE AGE** MAGAZINE

own microprogramming and machine architecture design to make a usable system out of the parts.

ROLLING ROBOT QUERY


TO THE ESTEEMED EDITORS OF BYTE:

WHAT CAN YOU TELL ME ABOUT A BOOK CALLED 'BUILD YOUR OWN WORKING ROBOT'
by DAVID C HEISERMAN
published by TAB BOOKS
Blue Ridge Summit, PA.

?

I NEED A PROJECT FOR MY INDEPENDENT STUDY IN MICROPROCESSORS NEXT SEMESTER.

I'll be SATISFIED WITH A ROBOT THAT ROLLS, RATHER THAN WALKS, AND WILL BE BIG ENOUGH TO HOLD MY 8080 C.P.U.

Doug Rosenberg
WHIZ ROBOTICS
R.D. 2
GERMANTOWN, N.Y.
12526

See page 58.

THE ETYMOLOGY OF HAM?

In the October BYTE, page 20, you published a letter from Bryan Patterson. In Mr Helmers' reply, he stated "ham" came from the tendency to "ham it up" on the air, as in the usage of "ham actor." The term "hams" came from the fact that in the early days of amateur radio, the government was trying to eliminate the use of radios for hobby purposes, and the four operators of one of the first stations that led the fight against the government were named Harriet, Arthur, Mark and Stanley; thus, the term hams came from the first letters of their first names. It then evolved into its present meaning of a radio amateur.

George J Valaitis
647 Linkfield Rd
Watertown CT 06795

WANTED: COMPUTER PERSON

I have subscribed to your magazine since the first issue and find the world of microcomputers fascinating. As yet I haven't built anything or purchased any hardware. I am still undecided as to what to do, what to buy, what to build . . . but I would very much like to get into this.

Would you print this in your magazine in hopes that someone near to me with an operating computer will contact me? I would like to see one running and talk with someone as interested as I am who has some experience with his own equipment. I am a ham radio operator, WA2TOI, and can be reached almost any time through the Gloversville 10-70 repeater or Cobleskill 01-61 repeater.

Also, if there is anyone in my area (50 miles west of Albany NY) who might be willing to help me get into this, I would appreciate a phone call or card.

Thank you for your interest and cooperation.

Jim Faliveno
Caroga Lake Stage
Gloversville NY 12078
(518) 725-2012, evenings

THE PERSONAL COMPUTING DIRECTORY, ANYONE? (AND OTHER TOPICS)

Allow me to congratulate you on your excellent journal. It's the only publication I've ever anxiously waited for every month.

I have a proposal: Someone like yourselves should publish a directory of computer hobbyists. You might call it something like Register 77. The directory would list anyone owning, having access to, or just interested in computers. It might also list clubs. The directory could be organized so that one could "key" on a person's name, address or microprocessor. Each name entry could contain information about the person. The following is an example of the information that might be contained, using myself as an example.

Name:
Larry J Leone
Age:
22
Address:
48 Second St
Masury OH 44438
Phone:
(216) 448-1252
Occupation:
Senior electrical engineering student, Youngstown State University
Languages:
FORTRAN, SNOBOL 4, BASIC, PL-1
System:
Constructing IM6100, 1 K RAM, access to IBM VM/370-145
Interest:
Artificial intelligence, robotics

I would imagine that many of the people that would submit their names to the directory would have access to a keypunch. If you set up a card format then they could send you the cards and *all* that you need to do is feed them into a card reader and let the computer do the work.

I don't believe there should be a wall between large and small computer systems, because they all work basically the same. I think some articles should be

about research work with large systems, how they are being used, and how they perform certain operations or functions. What I'm getting at is why reinvent the wheel when one can copy it; or, in this case a small system can mimic a large one. These kinds of articles should transmit stimulating ideas to the computer hobbyist.

Concerning the controversy over which micro is the best micro, I say any micro is better than no micro. I happen to be the proud owner of an IM6100, although if I had the money I would have a Z-80 with 64 K, but that's LIFE. I'm building my system from the micro up a little at a time; therefore it will most likely be incompatible with every other system, but that doesn't bother me either because it's all mine. I'm just happy to have something that was only a dream to anyone five or more years ago. I don't care how advanced a micro is; what more can one micro do over another in 4 K or less memory? At least I can get a mountain of cheap software made to run in 4 K for my system. Not everyone can make that statement. Remember a computer is only as good as its software.

A word concerning the Morse code articles of the October issue: I wouldn't be subscribing to BYTE if I wanted to read that type of article. I think the articles should have been spread through several issues or better yet sent to the appropriate magazine. I'm not criticizing the contents of the articles since I only glanced at them and quickly became uninterested in the topic. I'm sure you didn't mean this to happen but I know several friends who feel the same way. To help you from doing this, I am willing to attempt to write an article if you could suggest some topics that your readers want to read about.

At any rate, I do like the way you group similar articles into one issue. Keep up the good work.

Larry J Leone
48 Second St
Masury OH 44438

PS: Concerning the photo on page 96 of the December issue: I would guess it to be conductive foam used to ship ICs like CMOS.

Problem: How do you explain to the kids that are growing up in a world of digital clocks that something rotates clockwise? *[Good point ... Consult a physics text and talk about the right-hand rule and the cross product ... not exactly a perfect solution.]*

It turns out that someone is busily at work on just such a directory. His name is Richard Gardner, and his address is Resources, POB 134, Harvard Square, Cambridge MA 02138.

On a similar subject, PCC (Box 310, Menlo Park CA 94025) is preparing a Reference Book of Personal and Home Computing *for release this spring. Contact Dwight McCabe for information on this $4.95 book.* ■

DID YOU KNOW?

Did you know that when you buy the Midwest Scientific Instruments FD-8 Floppy Disk Memory System you get...

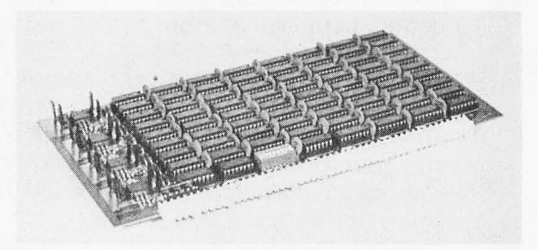
- the most complete disk operating system available anywhere for microcomputer systems
- disk routines that can be used easily by low level assembler language programs as well as high level BASIC programs
- program save and load with named, variable length, password protected files at three levels: object code files, source code files, and BASIC files
- a coresident assembler/editor that saves and loads source files from disk, assembles and stores object code on disk, and creates assembly listings on a high speed printer
- a full ANSI standard BASIC interpreter with program save and load, from disk as well as tape, with both random and sequential disk data files. BASIC also has multi-I/O port capability under software control
- and all you have to do to access any of these functions is type appropriate commands on your terminal.



Others promised but we delivered... a disk system that plugs in and runs.

FD-8 Floppy Disk System Kit **$1,150.00**
FD-8 System Wired & Tested **$1,395.00**

NEW!
8K RAM Boards for SWTP 6800



The new MSI 8K RAM Boards for the SWTP 6800 are available for immediate delivery. The RAM-68 uses low power 2102 memory with provision for a memory save battery pack in case of power failure.

8K RAM Board .... **$ 249.00**

Here is a list of MSI dealers who can show you our products in action

American Microprocessors
Equipment & Supply Corp.
Chicagoland Airport, P.O. Box 515
Prairie View, Illinois 60069
(312) 634-0076

Microcomputer Systems, Inc.
144 South Dale Mabry Avenue
Tampa, Florida 33609
(813) 879-4301

Micro Store
634 South Central Expressway
Richardson, Texas 75080
(214) 231-1096

Vanguard Systems Corporation
6812 San Pedro
San Antonio, Texas 78216
(512) 828-0553

A-Vid Electronics Company
1655 East 28th Street
Long Beach, California 90806
(213) 426-5526

Electronics for Yachting
1525 South East 16th Street
Ft. Lauderdale, Florida 33316
(305) 525-3478
TWX 510-955-9484

High Technology
1020 West Wilshire Blvd.
Oklahoma City, Oklahoma 73116
(405) 842-2021

Computer Workshop
174 Ifield Road
London England
SW10 9AG
01-373-8571

MSI is a dealer for Southwest Technical Products

Mastercharge & Bank Americard Orders Accepted

Send For New FREE Catalog

Before I forget...send me more information about the FD-8 Memory System and your new catalog.

Name ____________
Address ____________
City ____________
State ________ Zip ________

BY477

Midwest Scientific Instruments

220 West Cedar • Olathe, Kansas 66061
913/764-3273 • TWX 910 749 6403 (MSI OLAT)

Classified Ads

HELP: I need a schematic or any other information on a Friden 2018620, 260026 Assembly, PCB keyboard. I will pay for copying and a finder's fee. Bill Seeger, POB 2744, Keesler AFB MS 39534, (601) 435-4790.

FOR SALE: New Olivetti P-101 minicomputer. Easily programmed, permanent storage of programs on low cost magnetic cards, hard copy printout. Current list price, $1695. Will sacrifice for $995. Free delivery. Marc Seligman, 320 Bryn Mawr Av, Bala Cynwyd PA 19004.

WANTED: Low cost new or used terminal and/or acoustic coupler for remote use. Write Frank Scavo, 1811 Court St, Syracuse NY 13208.

FOR SALE: Dual floppy disk drive system. Shugart SA 3900 interface boards, two SA 800 disk drives, all power supplies in rack mount cabinet. Like new (used less than 10 hours). Operator's manual. First check for $1,800 to clear takes it. Also, new Remex RRK 2000 48 V paper tape reader, $240 and new Uchida PM-30N 48 V paper tape punch, $300. David Crellen, 4356 Vivaracho Ct, San Diego CA 92124.

WANTED: Centronics (101, 102) Diablo, Datapoint (2200, 1100), or HW, Hewlett-Packard, equivalent. NCR, Entrex, Univac, tubes, printers, disk, diskette drives. Wango, Wang, Kennedy, Ampex, Pertec, DEC, tape drives. In any condition, working units, scrap frames or just spare parts. Software Datapoint or HW, Hewlett-Packard, equivalent. Pat Snyder, POB 31, Franklin Square NY 11010.

STARTING NEW CORRESPONDENCE EXCHANGE CLUB: Would appreciate response from you on any useful project with a computer, visual or graphic, video camera interfacing to a computer. Any computer printer generated pictures or posters; Star Trek, Playboy, Snoopy, Cat, any figure or design on paper. Joe Penora, 920 First Av, Franklin Square NY 11010.

FOR SALE: RM 8080 PC board with documentation and edge connector. Includes 18 MHz crystal, 8224 clock, 8228 system controller, 8255 PIA and 2708 1 KX8 EPROM with system monitor, $120. Also, homebrew cabinet, sloping front panel with keyboard cutout 16 by 10 by 24 inches fashioned after BUD Comp-U-Cab series with SWTPC 6800 power supply mounted inside, $50. W T Walters, 2820 N Newell Rd, Oak Harbor WA 98277.

HELP: Need schematic for automatic electric AE101A data set, used on Model 28 KSR on the TWX line. Will pay for copying or original. Also Viatron 21 keyboard and main unit in excellent condition for sale, $300. Tom French, 404 Leeward Landing, Roswell GA 30076, (404) 993-1509.

FOR SALE: Four Lambda regulated power supplies Model LB-701FM-OV. Input is 208 VAC 3 phase, output is 0-8 VDC at up to 300 AMPS. Asking $300 each. Harold W Norris, RFD #3, Milton VT 05468.

FOR SALE: 5 low power 4 K RAM boards assembled and tested by E E. I'm using 8 K boards now and no longer need these, so they're yours for $100 each. S Edelman, POB 91, Ithaca NY 14850, (607) 272-1279.

SWAP: Will exchange a March 1976 issue of BYTE in return for either the May 1976 or June 1976 issue in good condition. D Kay, 460 W Clapier St, Philadelphia PA 19144, (215) VI9-6648.

FOR SALE: Two new unused MFE 250B digital cassette transports (see BYTE #15, page 71) with data; call, best offer. Also have Remex papertape spooler with schematic, $25 plus shipping. *Bug-Book III,* $10. Processor Tech's 5 K BASIC manuals and tape, $10. Assembled Morrow BYTE standard board, $100. Gary Alevy, 48D Clintwood Ct, Rochester NY 14620, (716) 244-1133.

WANTED: Bell 103 series data sets, answer/originate modes; also information on real time clock for SWTP 6800. Richard Courtney, 903 25th St, Marion IA 52302 (319) 377-0616.

FOR SALE: Altair 8800, working, hardly used, ICs in sockets, flat cable to front panel, $440 or offer. BYTE Shop TV typewriter, new, working. $110 or best offer. Dennis Feucht, 2015 SE 51st Av, Portland OR 97215 (503) 233-2648 or (503) 644-0161 ext 6255.

FOR SALE: Service manual for Selectric IO writer. $28. R Tierney, 4944 Cass St #811, San Diego CA 92109.

FOR SALE: 2 Itel Dura 1041B IBM Selectric word processors have built-in paper tape punch and paper tape reader coded in EBCDIC not ASCII, plugs for auxiliary reader and auxiliary punch. Word processor control box controls word, line, sentence, paragraph, margin and automatic editor. Have operating manuals and internal logic diagrams. Original cost about $7000. They have some bugs but work. $1000, each. G Hedden (714) 545-7888 evenings.

FOR SALE: Teletype ASR33 with special type wheel printing 12 characters per inch, 84 characters per line. Loaded; X on, off controls, etc. Service manuals included. Works well. $750. G Hedden (714) 545-7888 evenings.

WANTED: Communication adaptor complete for a Viatron system 21. David L Sammet, 817 Agathon Dr, Evansville IN 47712.

WANTED: MOS Technology 6502 resident assembler or source for same. Send any information to Steven D Swift, 15232 5th Av NE, Seattle WA 98155 or call (206) 774-2279 days, 364-3451 evenings.

FOR SALE: A 8080 disassembler, written in Altair 8 K BASIC, version 3.2. Dump your programs in assembler format. $2 for paper tape and instruction sheet, $.50 for listing. Chris Pettus, POB 611, Malibu CA 90265.

WANTED: May and June 1976 issues of BYTE magazine. Will pay $5 for each of these issues or will trade for December 1975, January, February or March 1976 issues of BYTE. Offer is open until I get one of each wanted issue. John Myers, 1556 36th St, Ogden UT 84403 (801) 399-2970.

HELP: I am collecting anecdotes, interesting ideas and descriptions of unusual projects concerning microcomputers (minis too) for a forthcoming book. If you're interested please send your article or description. It can be in very rough form; I'll help polish if necessary. You will be fully credited if material is used. Send all material to P C Davis, 17 1/2 Ackley Av, Johnson City NY 13790.

FOR SALE: ARP 2600 synthesizer, fully reconditioned, a dream at $1250. First call gets this beauty. Call Lee after 6 PM EST (201) 731-6958.

HELP: I need Issues 1, 2, 3, 6, 7, 8, 9, 10, and 12 of BYTE. Will pay any reasonable price. Send list to Alex Novickis, 8358 Park Av, Forestville CA 95436.

Readers who have equipment, software or other items to buy, sell or swap should send in a clearly typed notice to that effect. To be considered for publication, an advertisement should be clearly noncommercial, typed double spaced on plain white paper, and include complete name and address information. These notices are free of charge and will be printed one time only on a space available basis. Insertions should be limited to 100 words or less. Notices can be accepted from individuals or bona fide computer users clubs only. We can engage in no correspondence on these and your confirmation of placement is appearance in an issue of BYTE.

Please note that it may take three or four months for an ad to appear in the magazine.■

Microcomputer Glossary

David Price
3901 Victoria Ln
Midlothian VA 23113

In this article I have set down definitions of several computer words which one is likely to see in BYTE. I didn't want this to wind up being a full blown, unabridged dictionary, so I limited the list to words that I thought would be unfamiliar to most "outsiders." The computer industry has a fascinating tendency toward the use of abbreviations and acronyms. These words are printed in capital letters, and are spelled out within parentheses.

ACR (Audio Cassette Recorder) interface: Permits an ordinary tape recorder to be used to store digital data. It is frequently used as a low cost replacement for a magnetic tape unit.

AD (Analog Digital) conversion: An AD converter measures an input voltage, then outputs a digitally encoded number corresponding to the voltage.

Alphanumeric: The set of graphic symbols containing the alphabet (A-Z), the numbers (0-9), and special characters (such as punctuation).

Applications program: A program which does true problem solving, as opposed to an assembler, compiler, monitor or other systems program.

ASCII (American Standard Code for Information Interchange).

Assembler: Assemblers allow the programmer to write programs using easily memorized mnemonic instructions of an assembly language.

Assembly language: A low level symbolic programming language which was historically the first programming aid, and is the simplest type of language to implement.

BASIC (Beginner's All purpose Symbolic Instruction Code): BASIC is a widely used but simple beginner's high level language. Like all high level languages, it utilizes powerful statements, such as "10 PRINT A/B" instead of machine opcodes. BASIC is typically implemented as an interpreter, which runs more slowly than would be the case with a compiled computer language.

BCD: (Binary Coded Decimal).

Binary: The base-two number system. The digits are 0 and 1. They are used within a computer which represents them as the two states of an electrical circuit. The first five numbers (0-4) are 0,1,10,11,100.

Bit: This is a contraction used to name one binary digit.

Bug: A bug in a program is something which causes the program to malfunction. A programming error.

Byte: A group of bits. The most common byte size is 8 bits.

Chip: See integrated circuit.

COBOL (Common Business Oriented Language): A high level language heavily oriented to the use of files and record keeping. It replaces operation codes of assembly level languages with a set of powerful "verbs" resembling common English.

Compiler: A compiler accepts the statements of a high level language, then converts them to machine language.

Core memory: Many older computers utilize memories of Ferrite or Lithium Magnetic cores. A core memory doesn't "forget" data.

CPU (Central Processing Unit).

CRT (Cathode Ray Tube) display: A type of terminal which displays information on a television like cathode ray tube screen.

Data rate: The number of bits (or bytes, or words) of information transmitted over a communications channel per unit time.

Debugging: Removing bugs (programming errors) from the program.

Disk storage: A type of mass memory which stores data on a magnetically sensitive disk.

DOS (Disk Operating System): A computer system which uses disk storage. Also refers to special software routines for driving a disk system.

EBCDIC (Extended Binary Coded Decimal Interchange Code).

EROM (Erasable Read Only Memory): A ROM which can be erased and reprogrammed. The typical contemporary EROM is erased by exposure to ionizing radiation such as ultraviolet light.

Firmware: 1) Software which has been hard wired into the computer by storing it permanently on ROM, 2) Microprograms written to emulate a specific machine's instruction set.

XIMEDIA PRESENTS

The SOROC IQ120

CURSOR CONTROL. Forespace, backspace, up, down, new line, return, home, tab, PLUS ABSOLUTE CURSOR ADDRESSING.

TRANSMISSION MODES. Conversation (half and full Duplex) PLUS BLOCK MODE — transmit a page at a time.

FIELD PROTECTION. Any part of the display can be "protected" to prevent overtyping. Protected fields are displayed at reduced intensity.

EDITING. Clear screen, typeover, absolute cursor addressing, erase to end of page, erase to end of line, erase to end of field.

DISPLAY FORMAT. 24 lines by 80 characters (1,920 characters).

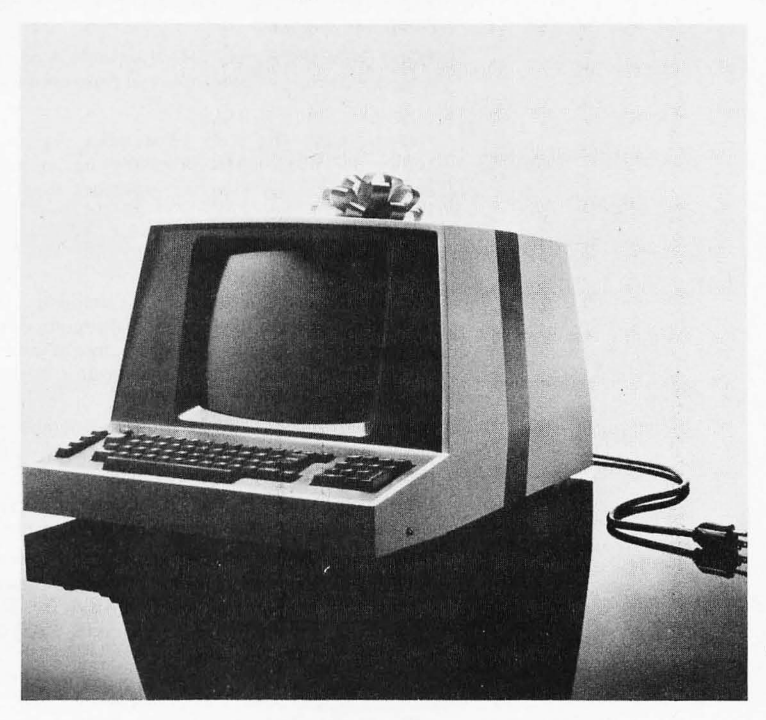
CHARACTER SET. 96 characters total. Upper and lower case ASCII.

KEYBOARD. 73 keys including numeric key pad.

REPEAT KEY. 15 cps repeat action.

DATA RATES. Thumbwheel selectable from 75 to 19,200 baud.

SCREEN. 12 inch rectangular CRT — P4 phosphor.



SPECIAL INTRODUCTORY PRICING

Kit $ 995.00 **Assembled $ 1,295.00**

(Price includes block mode, lower case and 24 line options.)

IMSAI + Z-80

XIMEDIA is offering a special price on an IMSAI I-8080 with a TDL ZPU Z-80 board instead of the standard 8080 cpu.

The system includes:
IMSAI I-8080 without cpu board. 22 slot mother board with 22 *pre-soldered* connectors and card guides; cooling fan.

Price:

Kit	Assembled
$ 999.00	$ 1,295.00

Selectric Terminals
$1,200.00

AT LAST!

An ASCII coded Selectric terminal with a RS232-C interface.

Call or write us today for details.

XIMEDIA OFFERS A FULL RANGE OF PRODUCTS FOR THE PERSONAL COMPUTER ENTHUSIAST AND THE SMALL SYSTEM DESIGNER. LET US QUOTE ON ALL YOUR HARDWARE AND SOFTWARE NEEDS.

OUR RETAIL STORE — THE COMPUTERIST™ — IS NOW OPEN IN SAN FRANCISCO. CALL US FOR DIRECTIONS.

XIMEDIA

1290 24th Avenue • San Francisco, CA 94122

(415) 566-7472. Please call collect!

California residents add 6% sales tax. Please allow 3 weeks for delivery.

Floppy disk: Disk storage which makes use of flexible disks made of a material similar to magnetic tape. Floppy disk units are slower and hold less data then hard surface disks.

Flowchart: A diagram representing the logic of a computer program.

FORTRAN (FORmula TRANslation): A computer language useful in computational programming.

Hexadecimal: The base 16 number system. It is commonly used when programming in machine language. (FD is easier to remember than "1111 1101".)

IC (Integrated Circuit).

Instruction set: The repertoire of the instructions a given machine can execute. A major component of a computer's design. The programs of one computer can't run on another computer if the two instruction sets aren't compatible.

Integrated circuit: A technique whereby many electronic components can be integrated and mass produced on a single chip of silicon.

Interface: A device which allows one type of electronic device to communicate with another.

Interpreter: An interpreter accepts and executes a high level language program. Unlike a compiler, however, it executes programs on the fly, instead of producing machine language code. BASIC is a common interpretive language.

LED (Light Emitting Diode): LED displays are often used as digital output devices because of their low weight, cost, and size.

Line printer: A high speed printer connected to a computer can print data at rates often exceeding several hundred lines per minute (LPM). This is achieved by printing a whole line at a time.

Looping: A programming technique used when a single portion of a program is to be repeated many times.

LSI (Large Scale Integration): The most complex form of an integrated circuit is made using LSI techniques.

Machine language: The lowest level of programming. This is the only language the computer can understand without the assistance of an assembler, compiler or interpreter.

Magnetic tape: Mass storage of data on magnetically sensitive tape, similar to that found in home tape recorders.

Microprocessor: A microprocessor consists of all the electronics of a minicomputer miniaturized onto a single IC chip.

Mnemonic: A short word which stands for another word or phase, designed so that it is easy to remember (ie: "Jump to subroutine" might be abbreviated "JSR").

Modem (modulator-demodulator): Converts audio sounds into digital form, and vice versa.

MPU (Micro Processor Unit): Synonym for CPU chip.

Octal: The base 8 number system. It is commonly used when programming in machine language ("375" is easier to remember than "011111101").

Opcode (OPeration CODE): A bit pattern which specifies a machine operation to a computer's central processor. Often listed in hexadecimal or octal notation.

Paper tape: A long strip of paper punched in a pattern decodable by the computer.

Parallel IO: When input and output is sent or received with each bit having a single wire, the transmission is called parallel. Contrast this to serial IO.

Peripheral: The devices which perform detailed interfacing operations for the computer. These include terminals, mass storage and the like.

Program: A set of instructions which instruct the computer to do a certain task.

Programming: To write a computer program.

PROM (Programmable Read Only Memory): See "ROM".

Queue: A "waiting line" of tasks or data.

RAM (Random Access Memory): Generic term for microcomputer system memory.

ROM (Read Only Memory): Memory whose data cannot be destroyed.

RS 232: An electronic interface standard for serial data communications.

Serial IO: A method of transmission in which bits are sent one by one.

Simulation: A computer program which models some system, typically using mathematical techniques.

Software: Computer programs.

Star Trek: A computer game based on the TV show "Star Trek." It involves destroying "Klingons" with an array of sophisticated weapons. Also referred to as "Space War."

Statement: A single computer instruction, within a computer program.

Subroutine: A subprogram within a larger program.

Terminal: A device for communicating with a computer using a keyboard and an alphanumeric printer or CRT display.

Teletype: A terminal which prints information on paper, manufactured by Teletype Corporation.

Timesharing: A technique whereby a single computer can support several terminals simultaneously.

TTL (Transistor-Transistor-Logic).

TTY (Teletype).

TVT (TeleVision Typewriter): An *inexpensive* form of video CRT display.

UART (Universal Asynchronous Receiver Transmitter): converts parallel data into serial form, or vice versa.

Word: A basic unit of computer memory.■



An Approach to High Level Languages for Small Systems

Donald J Stavely
1706 Banyan Dr
Fort Collins CO 80521

The prospect of high level languages for the computer hobbyist is exciting, though foreboding. Setting out to write a compiler for a microcomputer is in many ways more daring than taking on a compiler for a full-blown computer, since the "distance to be traveled" from source to machine language is greater. The large machine programmer has a vastly richer instruction set to work with. Also, the data elements of the source language (ie: integer, real number, etc) can often be directly implemented as single words on a large machine. For example, the FORTRAN expression REAL=REAL*PI translates to a mere handful of IBM 360 instructions, compared to perhaps a pageful of 8080 or 6800 instructions. The instruction set and word length not only make the final object code simpler, they also make the translation process (ie: the compiler program) simpler. If software complexity doesn't scare the small system user, then memory size ought to. Both the storage for the compiler and for the compiled program is much greater for a small and simple processor.

The reasonable approach for a small machine given these obstacles is to start by writing a series of "system subroutines" which will perform the complex operations done in a few instructions on a large machine. A "floating point package" is a prime example. Calls to floating point routines add structure, simplicity and compactness to compiled code. Also, scanning, search and block move routines simplify and shorten the compiler itself.

In essence, what we have done in this approach is to augment the puny instruction set and word length of our machine with subroutines. Once written and debugged, these routines "bootstrap" us up to a higher level of programming. We can now think in terms of "scanning for the next blank" instead of "load byte, compare to blank, conditional jump, increment pointer, etc, etc." We think "floating point add" instead of all that code to match the exponent, shift the multiple byte mantissa, add it, then normalize it back.

Now, armed with a few kilobytes of system subroutines, the compiler output becomes mostly a series of calls to these subroutines, rather than directly executable code. Think of this situation as if the compiler is no longer outputting true machine level code for your 8080 or 6800. It is outputting code for a more sophisticated, higher

About the Author

Donald J Stavely is an employee of the Hewlett-Packard Company, Ft Collins Division. He is a graduate of the University of Michigan, with a master's degree in Computer Engineering. In the letter accompanying this opinion, Don writes:

> I intended this piece for the new "Technical Forum" section, which I think is a great addition to BYTE. The "Technical Forum" will encourage people (like me) who have something to say, but who don't claim to be experts and who don't want to take on a full article with its implied burden of proof.

The letter and the original copy of this "Technical Forum" entry demonstrate an excellent application of a people oriented computer: Don typed the article using a text editing and formatting program on a 9830 calculator and a 9871 printer, both products of Hewlett-Packard. The 9830 is called a "desk top calculator" but is in reality a general purpose computer with some biases towards calculation built into its high level language.

level machine implemented through subroutine calls. What has been gained by this definition of a software machine design? First, the compiler is simpler because the level of the outputted code is one step higher than the real machine. Second, the object program code is much shorter since each call replaces many machine instructions. What have we lost? While the code is compact, it requires the few kilobytes of system subroutines in order to run. In terms of execution speed, we have sacrificed the overhead of the subroutine setups, calls and returns.

Let us now take this argument one step further. We have made the compiler easier and the compiled code more compact by making our machine appear more sophisticated than it really is through run time subroutines. What would we really like our machine to look like? A PDP-11? An IBM 360? Not necessarily like any conventional machine. We would like a machine with an instruction set tailored to compiling and running a high level language, with word lengths and data types appropriate to the task. We can define an "Abstract Machine" which we would like to have, and then make our microcomputer simulate this abstract machine by writing an interpreter program for it.

What would an interpreter program look like? As one would guess, there would be a main routine that would fetch the next abstract machine instruction and increment a simulated program counter. Based on the instruction, the main routine would branch to the appropriate execution routine. The abstract "opcode," rather than being an in line subroutine call, would require only a single byte — as an index into an instruction routine jump table.

In terms of architecture, we would like a machine that is at least partially stack oriented. Stack machines are well suited for processing high level languages for several reasons. First, stack code is very compact and storage efficient, since most operations have implied operands, ie: the top one, two or three elements on the stack. Second, the code generation portion of the compiler is greatly simplified since the parsing process directly implies stack operations. This applies not only to arithmetic expressions, but also to string expressions, logic expressions and nestable program constructs like IF-THEN-ELSEs, DO-WHILEs, etc. Third, stack allocation of memory is nice for recursive subroutines and block structured languages. The stack instructions would operate on the actual data types defined in the language. The abstract machine for an ALGOL-like language, for instance, might have stack elements which could be addresses, integers, floating point numbers, string descriptors or logical values.

Several things need to be said at this point. First, the notion of an intermediate level abstract machine is not new. More than a few compilers have been implemented this way. The IBM S/360 FORTRAN (G) compiler is written for a stack oriented machine called POP. PASCAL, an ALGOL-like language, also was written for an abstract machine. This concept is part of a broader class of methods to divide the compiling task into two parts: a Language Dependent Translator (LDT) and a Machine Dependent Translator (MDT). At what level and in what way this division is optimally made is the subject of current research.

Second, the idea of simulation or interpretation of intermediate code often brings to mind extremely slow and inefficient execution. This is certainly true when a conventional machine simulates another conventional machine. A 6800 simulation program of an 8080 machine is absurd. 95% of the execution time would be spent in the overhead of instruction fetching, decoding and setup. But a 6800 simulation of a 360 is not quite so ridiculous, since most of the time the 6800 would be grinding away at powerful arithmetic and logical routines. Defining an abstract machine whose "architecture" is suited to the source language and whose instruction set is powerful compared to the real machine insures that most of the execution time is spent doing useful work.

But now look at the advantages of the interpretive abstract machine for the amateur computing experimenter. First, as was said before, the code is extremely compact. A single byte instruction might do something like "take the floating point number at the top of the stack, convert it to an ASCII character string, and leave the descriptor in its place at the top of the stack." A fairly sophisticated program would take up little space, aside from the fixed overhead of the interpreter, string and numeric literals, and the value area. No matter how big memory chips get, it is safe to say that memory will always come dear to the computer hobbyist.

Second, an abstract machine is a step toward real machine independence and software portability. If you write an interpreter for your 6800 and your friend writes one for his 8080, abstract machine code will run well on both. Also, if you later upgrade to a 9900 or LSI-11, you don't have to junk all your software and start over — you just have to rewrite the abstract machine interpreter. The more powerful processor will make the interpreter shorter, simpler to write, and

faster at code execution. Machine independence should hold appeal to any serious programmer since it gives the ability to change and upgrade machines.

Third, it is a structured, "one step at a time" approach to software development. If your system now only has a few kilobytes of programmable memory, you can work on the interpreter, play with various abstract machine architectures, and write small programs in the abstract "machine language." Later, with a little more memory, a resident assembler written *in* abstract machine code *for* the abstract machine would be an interesting and useful project. Still later, with more memory, on to FORTRAN, ALGOL and APL!

If readers are interested, perhaps a basic abstract machine could be proposed and kicked around a little. Extensions to the instruction set and data types (like string or matrix operations) could easily be added later. Just how sophisticated and how language-specific the abstract machine ought to be is open for debate. I have played around with a simple one on my 6800 and am convinced it is the logical approach to high level languages on micros. It would seem to be a natural for those people who have big plans for their small systems.■

Continued from page 9

Listing 1, continued:

Address	Op	Operand	Label	Mnemonic	Commentary
007A	85	F9	FLY	STA INH	
007C	A9	02		LDA #02	} DECK:=02; [counter]
007E	85	E1		STA DECK	
0080	20	1F 1F	FLITE	JSR SCANDS	look for depressed key;
0083	F0	06		BEQ NOKEY	if no input go to NOKEY:
0085	20	6A 1F		JSR GETKEY	else go to GETKEY;
0088	20	91 00		JSR DOKEY	go to DOKEY;
008B	C6	E1	NOKEY	DEC DECK	DECK:=DECK-1;
008D	D0	F1		BNE FLITE	if DECK not equal to 0 go to FLITE;
008F	F0	D0		BEQ LINK	else go to LINK;
0091	C9	15	DOKEY	CMP #15	A:=fuel mode?;
0093	D0	03		BNE NALT	if not fuel mode go to NALT;
0095	85	EE		STA MODE	else MODE:= fuel mode;
0097	60			RTS	return;
0098	C9	10	NALT	CMP #10	A:=altitude mode?;
009A	D0	05		BNE NAL2	if not go to NAL2;
009C	A9	00		LDA #00	else mode:=altitude mode;
009E	85	EE		STA MODE	MODE:=A;
00A0	60		RET1	RTS	return;
00A1	10	FD	NAL2	BPL RET1	return; [illegal mode]
00A3	AA			TAX	else X:=A;
00A4	A5	EA		LDA THRUST	A:=THRUST;
00A6	F0	F8		BEQ RET1	if thrust:=0 go to RET1;
00A8	86	EA		STX THRUST	else THRUST:=X;
00AA	A5	EA	THRSET	LDA THRUST	A:=THRUST;
00AC	38			SEC	set carry;
00AD	E9	05		SBC #05	THRUST:=THRUST - 05;
00AF	85	E9		STA TH2+1	TH2+1:=THRUST;
00B1	A9	00		LDA #00	} A:=00;
00B3	E9	00		SBC #00	
00B5	85	E8		STA TH2	TH2:=00;
00B7	60			RTS	return;
00B8	45		INIT		} [initial height]
00B9	00				
00BA	00				
00BB	99				} [initial speed]
00BC	80				
00BD	00				
00BE	99				} [initial acceleration]
00BF	98				
00C0	02				[initial thrust]
00C1	08				} [initial fuel]
00C2	00				
00C3	00				
00C4	00				[mode]

until we realize that both the altitude and the fuel gauge will probably go right past the zero mark, jumping directly from a positive to a negative value; so a zero test is out. Instead, we take action the instant the number goes negative, restoring it to zero and then taking whatever other action is called for.

Lighting the Display

The display is quite straightforward; in fact, the KIM-1 monitor program has a subroutine to do the job.

Depending on the display mode flag, all we need to do is to move altitude or fuel to the display area, together with rate of descent. Then we call the subroutine to transfer it to the LEDs.

Of course, we must remember to drop the last two digits from the displayed values (0.01 of units, remember?) and to negate the rate of descent, where necessary, so that it shows as a positive number.

Detecting Input

The KIM-1 monitor subroutine that lights the display gives us a free bonus: It also tells us whether or not a key is depressed on the keyboard. To find out which key, we must call another subroutine in the monitor program.

If we discover that the user has input a thrust command, buttons 1 to 9, we first check to see that the motor is on and that we have fuel. Then we set the thrust, and also calculate the acceleration as thrust minus 5, where 5 represents the force of gravity.

The two other legal keys, A and F, set the display mode to altitude or fuel. The program sets a memory location which will be tested by the display routine.

The program doesn't need to worry about when a button is released. Although the question can be quite important for programs that must distinguish between, say, 9 and 99 on the input, the lunar lander doesn't really care. If you leave your finger on the button, it will keep on setting the thrust over and over to the same value, without affecting the flight.

Coming Down

The program doesn't stop. If you run out of fuel, you will watch yourself freefall to the surface. When you land, with or without fuel, your rate of descent freezes so that you can see how hard you landed.

It would be easy to have the display change after you land, to show words such as "SAFE" or "DEAD." The KIM-1 display is segment driven so that you can easily produce special combinations.

The novice astronaut who would like to try his or her hand at flying this, or other, craft should keep the following rules in mind:

1. Always conserve fuel at the beginning by reducing power to minimum thrust.
2. Don't let your rate of descent get excessively high; with my program, it's wise to steady up with a thrust value of 5 when your speed gets over 90 feet per second.
3. As you get to lower altitudes, try to balance your altitude against your rate of descent. At 1000 feet, a rate of descent of 500 feet per second will bring you down in 20 seconds, which is reasonable. Keep that sort of balance.■

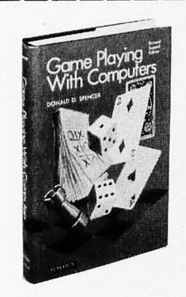
Continued from page 6

pages 106-107 of the Ballantine Books paperback edition:

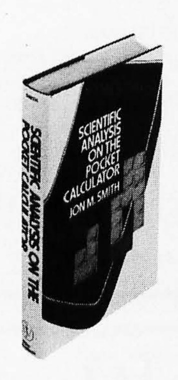
> *The 'Sec was the standard size of all such units, determined by what could fit comfortably in the normal human hand. At a quick glance, it did not differ greatly from one of the small electronic calculators that had started coming into general use in the late twentieth century. It was, however, infinitely more versatile, and Duncan could not imagine how life would be possible without it.*
>
> *Because of the finite size of clumsy human fingers, it had no more controls than its ancestors of three centuries earlier. There were fifty neat little studs; each, however, had a virtually unlimited number of functions, according to the mode of operation—for the character visible on each stud changed according to the mode. Thus on ALPHANUMERIC, twenty-six of the studs bore the letters of the alphabet, while ten showed the digits zero to nine. On MATH, the letters disappeared from the alphabetical studs and were replaced by X +, ÷, —, =, and all the standard mathematical functions.*
>
> *Another mode was DICTIONARY. The 'Sec stored over a hundred thousand words, whose three-line definitions could be displayed on the bright little screen, steadily rolling over page by page if desired. CLOCK and CALENDAR also used the screen for display, but for dealing with vast amounts of information it was desirable to link the 'Sec to the much larger screen of a standard Comsole. This could be done through the unit's optical interface—a tiny Transmit-Receive bull's-eye operating in the near ultraviolet. As long as this lens was in visual range of the corresponding sensor on a Comsole, the two units could happily exchange information at the rate of megabits per second. Thus when the 'Sec's own internal memory was saturated, its contents could be dumped into a larger store for permanent keeping; or, conversely, it could be loaded up through the optical link with any special data required for a particular job.*

The Minisec system's characteristics as envisaged by this passage are not really so

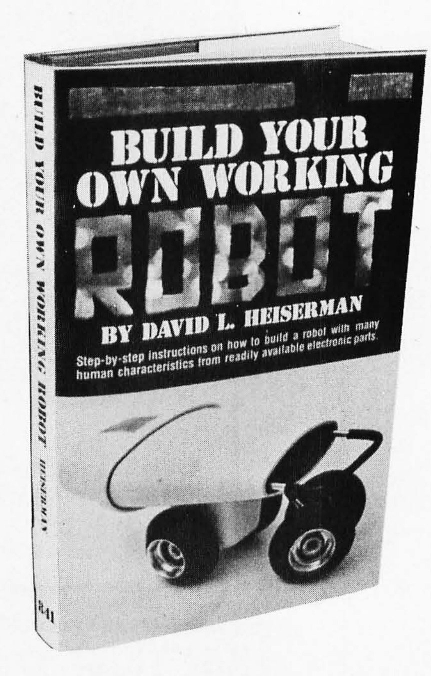
BITS™



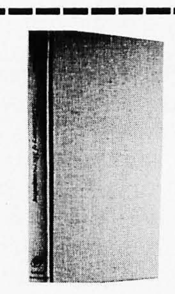
___**Game Playing With Computers** by Donald D Spencer, published by Hayden. What does it mean to play games using a computer? Read this book to get an introduction into numerous recreational uses of the computer to program and play mathematical and logical games. Topics include numerous mathematical problems, casino games, board games, unusual gambling games, and miscellaneous logic games. Numerous BASIC language programs and listings are included to show details. $16.95.



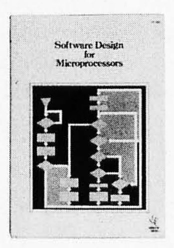
___**Scientific Analysis on the Pocket Calculator** by Jon M Smith, published by John Wiley & Sons. This book is another in a set of source books for mathematical analysis using the contemporary products of technology. It is oriented to the pocket calculator, yet it will provide you with algorithms and methods useful with any personal computer which implements the scientific and analytical functions found on a good pocket calculator. For a more complete description, see the book review on page 120 of the December 1976 BYTE; or order its 392 pages of detailed technical information and review its use for yourself. $13.75.



___**Build Your Own Working Robot** by David L Heiserman, published by Tab Books. This book will not tell you how to build Robbie, the robot of **Forbidden Planet**, or a classical android of science fiction. What it will introduce you to is the problems of making a robot mobile device called Buster III, using pre-microprocessor TTL integrated circuits for all logic functions. It is a must book for background reading, but much of the logic can be extremely simplified using today's microprocessor technology. Use this book as a first look at these problems from which you can build further and more elaborate solutions. Softbound, $5.95.



___**A Dictionary of Microcomputing** by Philip E Burton. In the opinion of BYTE's editor, Carl Helmers, "This is one of the best *designed* and executed dictionaries of computer related terms yet seen on the market. It is of particular relevance to those individuals who want a good general reference to numerous technical terms, broadly covering hardware and software fields as currently practiced." This new hardbound edition is part of the Garland Reference Library of Science and Technology. $12.50.



___**Software Design for Microprocessors.** This stand alone guide to microprocessors has been designed by the people at Texas Instruments to convey knowledge to the first time user of microprocessors. This excellent source book of computer concepts begins with an outline of the basic principles of the general purpose computer, its machine architecture, software, and methods of addressing. It proceeds to discuss how to build software, what is involved in documenting what you've done once you've done it, the mechanics of programming, and specific examples using the TI TMS-1000, TMS-8080, TMS-9900 and SBP0400 designs. You'll find a thick hardcover textbook filled with over 370 pages of useful information including a comprehensive glossary of microprocessor terminology, among several other detailed appendices. $12.95.

Send to:

BITS, Inc
70 Main St
Peterborough NH 03458

Total for all books checked $ ________
Postage, 25 cents per book for ________ books $ ________
Grand Total $ ________

Check payment method:
___My check is enclosed
___Bill my MC No. ____________________ Exp. date ________
___Bill my BAC No. ____________________ Exp. date ________

Name ____________________
Address ____________________
City ____________ State ______ Zip ______
Signature ____________________

You may photocopy this page if you wish to leave your BYTE intact. Please allow six weeks for delivery.

much science fiction. Aside from the size constraint and memory capacity, the functions of the Minisec as described can be accomplished by a well designed contemporary personal computer system. The characteristic of fitting into the size of a hand held object is not even necessarily a valid engineering goal for such a powerful device. Why?

Have you ever considered the problem of typing text passages into a keyboard the size of a calculator? Aside from the frequency and positioning idiosyncrazies of the standard Roman alphabet typewriters we all use (which could be solved by training with the Dvorak system), the design of the text keyboards is well adapted to the problem of quickly typing text. I know that personally from the immense amounts of keystroking I do, and anyone who works with computers at a software level will have similar experiences. A major portion of the information processing operations performed with the personal computer are now and will always be accomplished using such text keyboards, at least for the foreseeable future, barring invention of really efficient voice parsing techniques. Thus the physical size constraint is now and will probably remain at the level of the size of a text keyboard's physical layout, hardly a hand held size, but certainly a lap-perched device.

Redefining of key interpretations is already accomplished automatically when a command language is used with symbolic commands typed as text strings. However, having a changeable command keyboard such as Clarke describes is a neat idea. I ran into a specific case of such a keyboard when I worked for a short time on an experimental digital avionics system design which employed projection displays to define different legends for key positions depending upon mode of operation during an aircraft's mission. It is a nifty concept, but a bit expensive at present. The model of the Minisec keyboard is a very sensible extrapolation of liquid crystal style display technologies coupled with switchless capacitive touch contact zones.

Forgetting the physical characteristics of the implementation, the logical characteristics of multiple modes of operation are quite realizable with present day personal computers, although not always in a prepackaged form as described by Clarke. For example, the whole purpose of the ALPHANUMERIC mode is to allow text entry and command language style operations, if it is meaningful to have such a mode at all. The ASCII keyboard and the text editors applicable to personal computers of today

BITS™ Presents:



Please note: Although our ads will change month to month, with few exceptions all previous books advertised are available through BITS. A complete catalog will be sent to you with your order. . . . FWR

_____**TV Typewriter Cookbook** by Don Lancaster. A complete guide to low cost television display of alphanumeric data, several chapters of which were published ahead of the book in early issues of BYTE magazine. $9.95

_____**Digital Logic Circuits** by Sol Libes. An invaluable tutorial background volume on digital logic, arithmetic, IO concepts and interfacing to analog devices; written by one of the founders of the Amateur Computer Group of NJ. This book acquaints the reader with much of the terminology and background concepts of digital hardware. $5.98.

_____**Modern Operational Circuit Design** by John L Smith. An absolutely essential introduction to the use and application of operational amplifier systems. The book contains both theoretical background information and practical circuit suggestions which can be used to advantage by the experimenter. $22.00 (hardbound).

_____**Electronic Projects for Musicians** by Craig Anderton. "Even if you know nothing about electronics, you can build preamps, ring modulators, mixers, tone controls, miniampls, fuzzes and a dozen other inexpensive musical projects." Furthermore, even if you couldn't care less about about musical applications, you can use this book to gain familiarity with electronic parts, circuit diagrams and construction techniques, using the wealth of illustrations and tips found in an extended introductory chapter for the novice electronicker. $6.95.

_____**Active Filter Cookbook** by Don Lancaster. The chief chef of electronics Cookbooks concocts another gourmet appetizer. Run to this book when you need to find a starting point for the design of a filter for use in an electronic application. $14.95.

_____**Altair Design** developed by Ensor Holiday.

_____**More Altair Design** developed by Ensor Holiday.

_____**Altair Design 3** developed by Ensor Holiday.

Keep the loved ones busy while **you** use your system . . . give them any one or all of these computer generated **Altair Design** books to start them (or yourself) on endless hours of creative coloring. No other coloring books are quite like these unique books. $2.50 each.

_____**Artist and Computer** edited by Ruth Leavit. A visual treat, as you encounter reproductions of numerous works by computer oriented artists and read about these works in their own words. $4.95.

_____**Linear IC Principles, Experiments, and Projects** by Edward M Noll. From basic principles to complicated systems, from simple amplifier experiments to applications in radio, TV and control systems, this book can improve your knowledge of the way circuitry of the analog world really works. $8.95.

_____**Practical Solid-State Circuit Design** by Jerome E Oleksy. A self study course in the design of semiconductor circuits from the simple transistor to the complex operational amplifier. $5.95.

_____**101 BASIC Computer Games** by David Ahl. A treasure store of games and simulations, 101 all written in BASIC. It contains both a complete listing and a sample run, along with a descriptive write-up. 256 pages. $7.50.

_____**Projects in Sight, Sound & Sensation** by Mitchell Waite. Dedicated "to all space cowboys." Detailed theory and practice of seven fascinating amateur electronics projects, along with a complete and detailed appendix on how to make PC boards. $4.95.

_____**Security Electronics** by John E Cunningham. To catch a thief, apply liberal doses of ingenuity and a modicum of cleverness. Find out what's been tried in conventional alarm systems before you go off computerizing your home security system, though. $5.95, new second edition.

_____**Introduction to Biomedical Electronics** by Edward J Bukstein. What's been done in robot doctors? Nothing so far. But in terms of electronic aids to physicians and practices of health researchers, consult this background review of the field of biomedical electronics. $5.50.

_____**What to Do After You Hit Return of PCC's First Book of Computer Games.** This is PCC's first book of computer games, a compendium which includes descriptions of numerous games, and listings of 37 selected BASIC games. $8, new second edition.

Send to:

BITS, Inc
70 Main St
Peterborough NH 03458

master charge | BANKAMERICARD

Check payment method:
___My check is enclosed
___Bill my MC No. __________ Exp. date ______
___Bill my BAC No. __________ Exp. date ______

Name ____________________

Address ____________________

City __________ State _____ Zip _____

Signature ____________________

Total for all books checked $ ______
Postage, 25 cents per book for ______ books $ ______
Grand Total $ ______

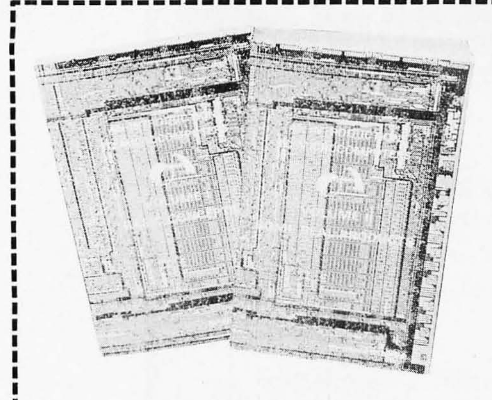
You may photocopy this page if you wish to leave your BYTE intact. Please allow six weeks for delivery.

provide this mode of operation. The MATH mode is simply extending the concept of a BASIC or APL or language X interpreter to allow its use with your personal computer to calculate programmed combinations of elementary and transcendental functions. DICTIONARY mode is a little bit more difficult, primarily because as of today we do not yet have the technology to inexpensively store (with random access) such a large vocabulary with associated definition materials. But, given a mass storage drive with 2 million or so bytes on line (for example, a 3M cartridge) and enough time to do the keystroking, a personal information system (a la dictionary) with a significant size can be built using equipment presently available on the market. The CLOCK and CALENDAR functions of the Minisec are so trivial to implement that it's amazing it isn't standard on all personal computers which are assumed to be plugged in 24 hours a day: It only takes dedication of an interrupt line driven by timing signals generated by frequency division from the crystal controlled clock of the processor. The interrupt routine updates the real time as stored in an appropriate programmable memory location or locations, scaling and converting to 24 hour time if desired.

As for connection to external larger information systems, the present day personal computer does not exactly use a fast light interface, but it certainly can use a modem and telephone link at 300 bps or so. Modems are available in kit or surplus form at quite reasonable prices today, and are likely to be a drop in the price bucket compared to the costs of the timesharing services themselves unless you just talk to your neighbor's computer.

The Future Comes Faster Than Anticipated

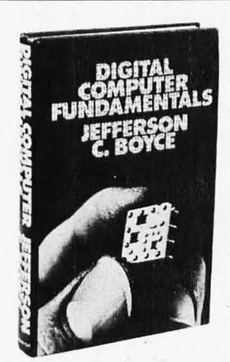
What this all boils down to is the conclusion that the design and function of Arthur C Clarke's conception of the ultimate personal computer is a lot closer to reality than he might expect. The actual dates he gives for the progress of technology into the Minisec stage I do not recall, but I certainly suspect that he put it further into the future than the present day and the next few years. But, then, one purpose of science fiction is to explore technological scenarios given an author's defined set of outrageous premises. No one should believe that there is any prophecy about when the scenarios are played if ever. Rather, the ideas and speculations serve as inputs to the imaginations of the innovators of practical products as the technologies progress. Viewed in this light, Arthur C Clarke's concept of the Minisec



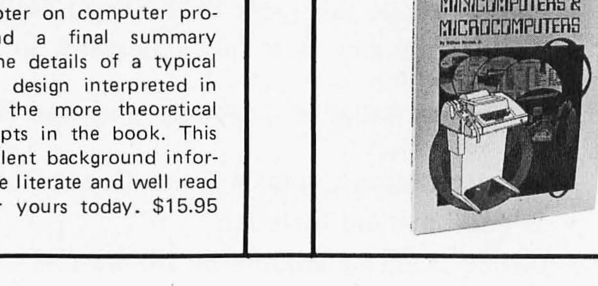
Adam Osborne's books **An Introduction to Microcomputers, Volumes 1 and 2,** are a concise compendium of the technical details of microprocessors at the component (engineering) level. These are the source books for the system designer who plans to employ the microprocessor, or the advanced homebrewer who wants a dash of customization not found in commercial products.

____ Volume 1 is subtitled "Basic Concepts." This is the book which presents a framework of ideas concerning the design and use of small computers implemented with LSI. Topics include definitions of the microcomputer, fundamental concepts of logic and numbering characteristics of instruction sets, etc. $7.50.

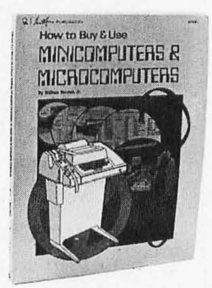
____ Volume 2 is a much thicker (895 page) detailed volume which complements the information in the first volume. This is the volume which fills in many of the details left out of the conceptual treatment in Volume 1. Here you'll find 19 detailed chapters on the engineering and logical specifications of products made by 16 different manufacturers, including in many cases reprints from the manufacturers' documentation as well as new materials provided by the author. Published in 1976, it even includes such processors as the MicroNOVA by Data General and the Texas Instruments TMS-9900 as well as the older 8 and 16 bit machines. Organization is by design type, and where parts of several manufacturers were intended for a given processor design such as the 8080, these are grouped into a single chapter. $12.50

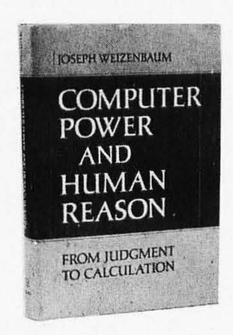


____ **Digital Computer Fundamentals** by Jefferson C Boyce. The way to a world of learning is through books. A great place to start, and to return from time to time, is the classical textbook. This new book from Prentice Hall is intended as just that. Topics covered include digital computer operation, basic computer circuits and concepts, Boolean algebra, implementing computer operations in hardware, communicating with the computer and related issues of coding schemes, detailed discussions of the control section, memory functions, arithmetic and logic functions, input and output functions of a classical computer, a chapter on computer programming and a final summary chapter on the details of a typical minicomputer design interpreted in the light of the more theoretical general concepts in the book. This book is excellent background information for the literate and well read hacker. Order yours today. $15.95 hardbound.



____ **How to Buy and Use Minicomputers and Microcomputers** by William Barden. People have often asked us where to turn to get an introductory book about computers for personal use. One excellent place to start is **How to Buy and Use Minicomputers and Microcomputers,** William Barden Jr's instant summary of the small computer revolution, published by Howard Sams in mid-1976. This is one of the first books of the "general introduction to computers" genre to be published with an emphasis towards the small computer and personal computing as it is being practiced these days. the book, written for the novice as well as the expert, surveys the technical details of the field in nine chapters and 10 appendices. This book is light (but essential) reading for the experienced computer person, and worthy of serious, concentrated perusal by the novice. $9.95.





____ **Computer Power And Human Reason** by Joseph Weizenbaum. This book is one which should be purchased or read for several reasons. If you're presently a programmer by trade or skill, you'll see a philosophy of computer use and abuse propounded. It's genuinely interesting, and definitely provocative if you reference the storm of letters, counter letters and counter counter letters which this book produced in the Association for Computing Machinery's *SIGART* newsletters during 1976. If you're a novice to the field, the tutorial and explanatory chapters of this book, which are aimed at the layman, will serve as an excellent background source which is also eminently readable. This includes an excellent and low level explanation of what an algorithm is, and how computers go about executing effective algorithms. $5.95 softbound.

Send to:

BITS, Inc
70 Main St
Peterborough NH 03458

Total for all books checked $ ________
Postage, 25 cents per book for ________ **books** $ ________
Grand Total $ ________

Check payment method:

___ My check is enclosed
___ Bill my MC No. ____________ Exp. date ______
___ Bill my BAC No. ____________ Exp. date ______

Name ____________

Address ____________

City ________ State ____ Zip ____

Signature ____________

You may photocopy this page if you wish to leave your BYTE intact. Please allow six weeks for delivery.

personal computer and its uses throughout the book *Imperial Earth* is a most inspiring input which designers of personal computers would do well to explore.■

A CHALLENGE . . . The Logical Squelch, a Noncommercial Product

Occasional, functional specifications of neat ideas can be fun. Here is one which was dreamed up by yours truly and Judy Havey while preparing the April issue. We both love to listen to classical music FM radio stations while working, principally WGBH and WCRB out of Boston MA. What we were talking about was the fact that while we certainly appreciate the fact that advertising supports the commercial radio station WCRB, we sometimes get just a bit tired of hearing the same commercial or announcement over and over again. So, I said to myself, wouldn't it be a neat hack to develop an audio input processor backed up by an AI algorithm which has the following characteristics:

- Continuous digitization of the signal and characterization of the past "n" milliseconds by a set of numerical constants reflecting the signal being digitized.
- A manual switch input used for "catching" various unwanted signals and storing the empirical parameters for the previous time period (goal: perhaps 16 to 128 bytes characterizing a segment of the signal with a high probability of uniqueness).
- A monitoring algorithm which seeks a match between stored signal parameter sets and the incoming pattern, perhaps with a "criticality" parameter which adjusts a fudge factor tolerance on the match. When a match is found, the "squelch" signal is given to the output level control peripheral.
- A level control peripheral which is either full "on" or in a squelched state, with a timing oneshot used to stretch the squelch period over an adjustable time of a half to two minutes or so.

In this simplest conception, no attempt is made at fancy word recognition, or recognition of voice characteristics of announcers. Such embellishments would of course be most desirable, but who knows what is possible until it is tried? Who'll be the first to build this logical squelch device and write it up as an article for perusal of our other readers? . . . CH

BYTE's Bits

Attention Education Users

The fifteenth annual convention of the Association for Educational Data Systems will be held April 25-29 1977 at the Green Oaks Inn, Fort Worth TX. For further information contact Alton R Goddard, publicity chairman, at AEDS National Headquarters, 1201 16th St NW, Washington DC 20036, (202) 833-4100.■

Catalog of Computer Supplies Available

A new catalog from Digital Equipment Corporation, "Supplies and Accessories," pictures and describes 200 replacement and enhancement items for use with Digital peripheral equipment, including media for disk and magnetic tape drives; paper supplies, ribbons and baskets for printers, plotters, terminals, paper tape and card readers; terminal stands, chairs, tables and other convenience accessories. The 56 page color catalog includes order blanks and is available from Communications Services, 444 Whitney St, Northboro MA 01732.■

PL/I

1977

JANUARY

S	M	T	W	T	F	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

FEBRUARY

S	M	T	W	T	F	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28					

MARCH

S	M	T	W	T	F	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

APRIL

S	M	T	W	T	F	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

MAY

S	M	T	W	T	F	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

JUNE

S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

JULY

S	M	T	W	T	F	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

AUGUST

S	M	T	W	T	F	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

SEPTEMBER

S	M	T	W	T	F	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

OCTOBER

S	M	T	W	T	F	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

NOVEMBER

S	M	T	W	T	F	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

DECEMBER

S	M	T	W	T	F	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

the

UNCOBOL

This Bit was supplied by Chuck Adams at Texas A&M University, College Station TX.

The Grapevine's Latest Goblet of Divine Juice

According to the volcano of rumors and other premature enthusiasms, Jim Warren of *Dr Dobb's Journal of Computer Calisthenics and Orthodontia* (PCC Box 310, Menlo Park CA) in a public utterance at a Homebrew Computer Club meeting January 19, there are some goodies in the works from Zilog. Dr Dobb himself (alias Jim Warren) says that reliable reports indicate the Z-200 is Zilog's next product, a high level language stack machine which directly executes PASCAL constructs as its machine level language. It remains to be seen whether this particular concoction is grounded in reality or the result of several stages of quite nonlinear amplification. ■

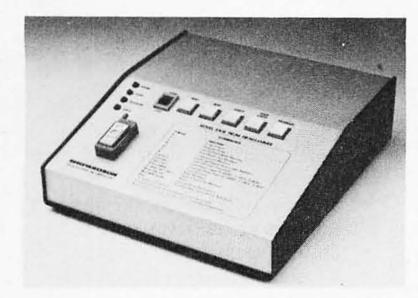
Two Computers in Every Home, Motherhood, Pizza, Apple Pie, et al



The latest word we hear is that a firm with the name Computer Shack, tied up in some nebulous way with the people who bring you another kind of electronics oriented Shack, is planning to double or quadruple the present number of computer stores in existence through a nationwide franchise program (see advertisements in BYTE, as well as an ad which has run in *Computerworld* several times). Here's one of the people responsible, Ed Faber, president of Computer Shack Inc, located at 14860 Wicks Blvd, San Leandro CA 94577.■



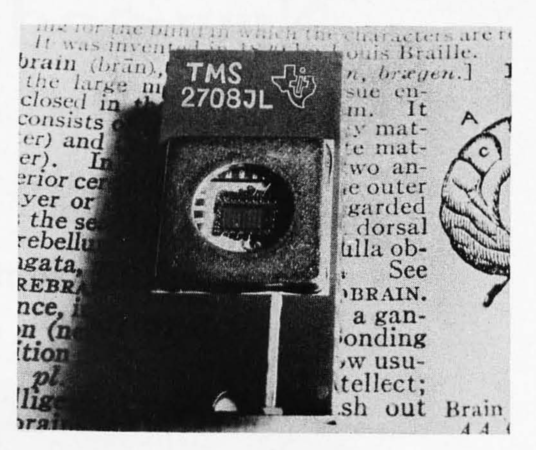
ROM Programs Beget ROM Programs . . .



This new programmer for 2704 and 2708 memory parts is an example of the creative use of a microprocessor to engineer a system which has features which would be difficult if not impossible to achieve previously. The system is a microprocessor controlled programmer which allows the user to create, edit and manipulate data in a buffer memory using an RS-232 or 20 mA current loop (Teletype) terminal interface. In a hypothetical computer store PROM programming center, this programmer with its interactive command sequences would be connected to a terminal or one of the demonstration computers of the store, so that customers could edit and try out programs before committing them to PROM. One can conceive of the retail store selling time on the programmer at X$ per hour, with X set to amortize the cost of this $850 device and its terminal in a reasonable time. And of course the industrial and commercial firms who design products with microprocessors will find this device a most useful piece of equipment for the microprocessor design and production facility. The company which makes this device is Shepardson Microsystems Inc, 20823 Stevens Creek Blvd, Bldg C4-H, Cupertino CA 95014. According to the press release, new products which include a multiple socket version of this programmer and additional programmers with similar features oriented towards bipolar PROM products are due soon.■

Finally, We'll Get a Low Enough Price for the 2708

Texas Instruments has recently announced its entry into the 2708 EROM market with the TMS2708JL part. This is an 8 bit by 1 K memory part with 450 ns access time, static operation and TTL compatibility, in a 24 pin package *which will plug right* into many microprocessor applications. We've got an article coming up in a future issue of BYTE on how to program these chips, which will be seen more and more frequently in personal computing circles over the next half year or so. The significance of the TI announcement is that it marks entry of alternate sources into a market which in the past has been characterized by high initial production prices. At BYTE, we paid nearly $100 for a 2708 several months ago (quantity one, from a distributor). As TI and all the rest of the new 2708 sources compete their way down the "learning curve," we can expect to see these prices drop if past history is any clue. Don't be surprised if the cost of a completely filling a Cromemco BYTESaver board with 2708s (8 K worth in 8 chips) has dropped to about $400 within one year. Locally programmable EROM parts are quite useful to have around for storing



frequently used programs; so the more vendors there are, the better the prospects for a reasonable price.■

From Sweden: LYS-16

Göran Peterson, of AB Aero-Telaw Atew, Box 125, S-642 00 Flen, Sweden, sends us a flier on a new small computer product which his company has introduced. The product is the LYS-16, designed by the Computer Society of the Linköping Technical High School in Sweden, and available either in kit form or as a finished product. The technical specifications of this product include:

- Memory address space for up to 64 K bytes.
- 256 byte miniature operating system including a 1200 bps serial interface for audio recorders.
- television interface (which obviously must be designed for the local power line frequencies).

The first deliveries are scheduled for the spring of 1977, and the price is listed

as 5900 Sw Crw (based on foreign exchange rates quoted in the December 15 1976 edition of the *Wall Street Journal*, this is equivalent to about $1357 in US currency). According to the flier, the company has about 180 employees and plans to offer a floppy disk option in the near future as well.■

Micro Cosmos PLEASE

Robert Tripp, publisher of *The Computerist*, has announced a package of software for the KIM-1 system which he's selling for $10. According to his announcement, the features of this package, distributed on a *KIM-1* cassette, include:

- timer routines with 1 ms accuracy.
- a clock program displaying hours, minutes and seconds.
- an adding machine program with six digit addition/subtraction.
- a decimal to hexadecimal (and vice versa) conversion routine.
- a drunk test program.
- and a much longer list of gimmicks and programs . . . .

He even claims to have a "high level language" as part of the package, and that the whole set of programs is designed to run in an unmodified KIM with minimum memory. The package is to be distributed with complete source listings, operating instructions for the demonstration programs, and instructions on how to write programs in the PLEASE high level form.

Write Micro Cosmos, 210 Daniel Webster Hwy S, South Nashua NH 03060.■

New Hobbyist System Supports Timesharing

The SIXTEEN/8, a new personal computer announced by Western Digital Corp and The Computer Mart of Orange County CA, is the first personal system to our knowledge which supports full scale timesharing with multiple terminals and active jobs. The system offers an instruction set similar to the Digital Equipment Corp LSI-11, but comes as two fully assembled cards which plug into an Altair or IMSAI backplane and permit the use of existing memory and peripherals. The price of the system with software is $1295, and the source code for all software is available on floppy disk for the cost of distribution.

The SIXTEEN/8 hardware, microcode and software were developed by Dick Wilcox, a Western Digital employee heavily involved in development of the LSI-11. The new machine's microcode implements essentially the PDP-11 instruction set as seen from the assembly language level, including the floating point option; but the instruction encoding is different and several PDP-11 instructions have been omitted in order to discourage the use of proprietary DEC software. A number of instructions have

been added which considerably enhance the power of the SIXTEEN/8: these include short immediate mode, multiple bit shift, and block move instructions; table indexed jump and subroutine call instructions; "load effective address"; and instructions to save and restore registers on the stack. Typical instruction execution times for the SIXTEEN/8 range from 3 to 10 µs.

To permit the use of existing 8 bit microprocessor memory and peripherals, the PDP-11 peripheral addressing structure was revamped to support 8080 style IO transfers, and external demultiplexing logic involving some 70 TTL chips was added to interface a 16 bit processor to an 8 bit bus. The SIXTEEN/8 also supports 16 bit memory, vectored interrupts, and multiple DMA ports.

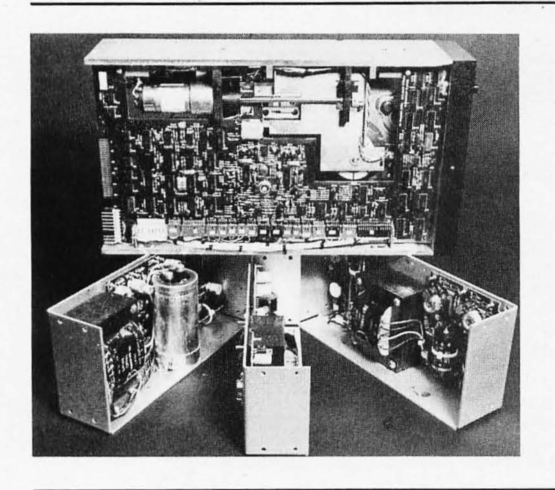
The software for the SIXTEEN/8 was developed by Dick Wilcox over a period of 4 1/2 years on a PDP-11/40 in his back bedroom and has been sold to a number of large commercial users. The basic operating system supports floppy disk space management and a multi-user file system; multiple interactive terminals, each of which may control several concurrently running jobs, and segmentation and overlay loading for user programs. Operating system calls support device independent IO for all programs and devices. User written drivers for special purpose devices are easily added to the system. The timesharing facilities do not, however, include virtual memory management or swapping to floppy disk; nor is memory protection provided at present.

Program development software includes a text editor, a macro assembler, a linkage editor which combines separately assembled programs, and a symbolic debugger which permits run time references to source code names. A powerful BASIC compiler (not quite finished at the time of this writing in mid-January) supports a wide range of BASIC language features, 11 digit floating-point precision, and interactive or batch execution with speeds estimated at 18-20 times that of MITS 8 K BASIC. An 8080 cross assembler is also provided with the system. Other software includes a line print spooler and a CRT based job status monitor. The entire source code occupies about two dozen fully loaded floppy disks.

Use of the system is likely to require substantial amounts of main memory. The basic operating system occupies 10 to 11 K bytes, and the BASIC compiler requires 12 K bytes; a practical system for a single user might well include 24 K to 32 K bytes of memory.

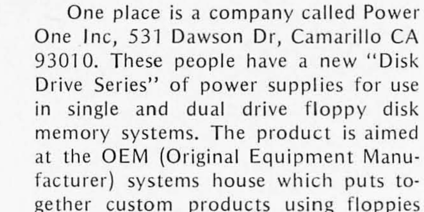
Although the features described for this system seem quite incredible, the SIXTEEN/8 was demonstrated recently to members of the Homebrew Computer Club in Santa Clara CA and appears to be a reality. Deliveries should be underway by the time this note is published. Also planned for later introduction is a complete package using 16 bit memory and a self-contained floppy disk drive,

controller, and power supply. Inquiries about the SIXTEEN/8 should be addressed to The Computer Mart, 625 W Katella, #10, Orange CA 92667, (714) 633-1222, or to your local Western Digital representative. **—Daniel Fylstra**



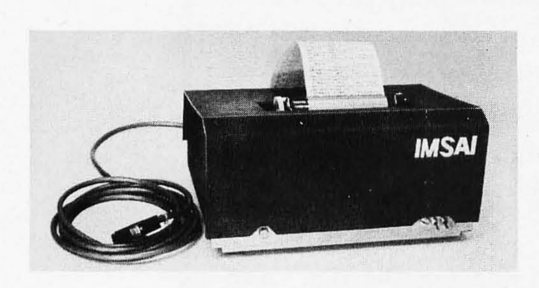
Attention Floppy Users . . . Here's a Power Supply Module for You

Suppose you decide to build a floppy interface, such as the one Ken Welles described in February BYTE. One of the factors you'll have to consider is the power supply for the floppy itself. This key component is not part of the price of the bare drive, but generally is included if you buy a commercial floppy interface package. If you have the right voltages and spare current capacity sitting around in some other supply of your system, there is no problem, but where do you go if you don't have a proper supply?



One place is a company called Power One Inc, 531 Dawson Dr, Camarillo CA 93010. These people have a new "Disk Drive Series" of power supplies for use in single and dual drive floppy disk memory systems. The product is aimed at the OEM (Original Equipment Manufacturer) systems house which puts together custom products using floppies and other subsystems.

These supplies will also prove to be a good item for computer stores to carry to augment independent handling of floppy disk component lines and interfaces. Retailers should find the 2 year factory warranty a useful selling point along with overload and overvoltage protection. For individuals dealing direct and prepaid, the 1 to 9 unit prices *range* from $69.95 to $120 depending upon model. If you're interested, write for the brochure on the disk drive series, Models CP-162, CP-205 and CP-206.■



The IMSAI Hard Copy Option

This new $399 (kit, or $549 assembled) printer interfaces to an 8 bit parallel output port. It is a selfcontained unit with case, cable, power supply, timing, control and character generation logic. The interface is through a parallel cable so it should be quite easy to install through any parallel interface board. The type font is a standard 64 character ASCII subset, with the option of printing double size characters for headings. The controller features automatic line wraparound so that output lines greater than 44 characters in length are printed on two or more lines with no software tricks. Since it is an impact printer it can be used with multiple copy forms using carbon or NCR paper. For further information on the IMSAI Printer, write to IMS Associates Inc, 14860 Wicks Blvd, San Leandro CA 94577.■

Computer Stores in Canada — How many are there?

Many stores have opened in Canada within the last year or so. Seven stores are known to us, and there are probably quite a few more which we do not know about. We would appreciate our readers telling us about stores not listed here.

Pacific Computer Store
4509-4511 Rupert St
Vancouver BC V5R 2J4
(604) 438-3281

The Computer Place
186 Queen St W
Toronto ONT M5V 1Z1
(416) 598-0262

Focus Scientific
160 Elgin St
Ottawa ONT K2P 2C4
(613) 236-7767

Computer Master Systems Ltd
69 Gloucester St
Toronto ONT M4Y 1LB
(416) 924-9789

First Canadian Computer Store Ltd
44 Elington Ave
Toronto ONT M4R 1A1
(416) 482-8080

Computer Hobby Shop
4812 16th St SW
Calgary ALBERTA T2T 4J5
(403) 243-6776

The Computer Shop
3515 18th St SW
Calgary ALBERTA
(403) 243-0301 ■

BYTE Subscriptions

Since our subscriptions have increased so greatly, we have decided to use a subscription service. This new service will make it possible to serve you better and faster.

According to theory, any systems transition is supposed to be smooth. However, if there should be some unexpected and unplanned problems, please bear with us during this transition period.

Please address all subscription correspondence to

BYTE Subscriptions
PO Box 361
Arlington MA 02174
(617) 646-4329

Clubs and Newsletters

Want to help your club reach out to the nearly 200 computer groups around the country and the world, share information, newsletters and valuable organizational experience? That's what this column is here for. We'd like nothing more than to help your group benefit from the attention it will receive from a writeup in a magazine that's widely read by people like you. People who are forming groups are invited to send details or questions. New clubs are especially welcome. We ask the more established to put us on their mailing lists. All correspondence should be addressed to Peter Travisano, BYTE, 70 Main St, Peterborough NH 03458.

Personal Systems – San Diego Computer Society

It's always a pleasure to see a job well done and the San Diego Computer Society's publication *Personal Systems* does its job well on a monthly basis. There's plenty here for the thoughtful hobbyist to read through, think about and experiment with. Dr Lance Leventhal's first rate series, "The Micro-Tower" is always a treat, and there are plenty of other goodies. A recent issue carried what looked to be an interesting program in F-8 machine code called Mastermind – a game of code breaking which will probably tantalize and challenge the best of us. Someone finally did it, the article many of us wanted to see, but were afraid to ask for, "The Complete and Utter Idiot's Guide to Computer Programming," a reprint from a recent issue of the Homebrew Computer Club's newsletter. This should save a lot of aggravation for people just breaking into computers.

Of interest to anyone with the ability to get there is the West Coast Computer Faire to be held April 15 through 17 1977 at the San Francisco Civic Center. According to *Personal Systems*, the Faire is expected to draw from 7000 to 10,000 people. There will be 50 to 100 tutorials, an exposition of homebrewed, exotic, and low cost systems, components, and you name it, along with commercial exhibits. In short, it promises to be an incredible experience for those interested in computers.

If you'd like to find out more about the San Diego Computer Society, drop a note to POB 9988, San Diego CA 92109.

Computer Ham? Check out AMRAD

The Amateur Radio Research and Development Corporation is by, of, and for those people who are the vanguard in their hobby, ham radio. Speeding up radio communications often means the employment of computers. AMRAD is making an attempt at organizing a regular newsletter which could conceivably be a boon to people of similar interests. If it's going to work, they'll need your support. Contact Amateur Radio Research and Development Corporation, 1524 Springvale Av, McLean VA 22101.

Trenton Computer Festival

Trenton NJ, the place where it all began for amateur computer festivals, is about to do it again on April 31 and May 1 1977 at Trenton State College, Rt 31, Trenton NJ. It looks like this is going to be a good sized fair with 30 speakers, up to 90 demonstrators and an outdoor flea market. Forums are planned on microcomputers for home, radio amateurs, education and medicine, consumer applications of microprocessors, computer music, robots, graphics, speech synthesis, establishing amateur computer standards and a computer club convention.

Sponsors include a number of groups: Amateur Computer Group of New Jersey, Philadelphia Area Computer Society, Trenton State College Digital Computer Society, Institute of Electrical and Electronic Engineers – Princeton Section and the Department of Engineering Technology, Trenton State College.

It is expected that attendance will exceed 3,000. There will be amateur computing contests and awards, a Saturday night awards banquet, programming copying service, forum audio cassette copying service, free bus service between hotels and motels and fest, activities for wives and kids, and free parking for up to 4,000 cars.

For more information write Trenton Computer Festival, Trenton State College, Trenton NJ 08625.

Société d'Informatique Amateur du Québec

Computer buffs in Quebec will be glad to hear of the formation of the first French speaking computer club, Société d'Informatique Amateur du Québec. Hobbyists from everywhere in the province are invited to get in touch with Ricardo Talbot at (418) 522-3349 during the day or (418)

522-4872 at night. Correspondence should be directed to him through IRISCO du Québec Inc, 376 du Roi, Suite 304, Québec PQ 2W6. SIAQ looks forward to relations with all clubs in Canada or the US willing to forward their newsletter. Bon chance.

South Florida Computer Group

We came across this ingenious little ditty in volume I #9 of the South Florida Computer Group's newsletter reprinted from "Florida Skip" and thought we'd pass it on to you.

Xvxn though my typwwritxr is an old modxl, it works quitx wxll xcxpt for onx kxy. Thxrx arx 46 kxys that function wxll xnough, but just onx kxy not working makxs thx diffxrxncx.

Somxtimxs it sxxms to mx that our group is somxwhat likx my typxwritxr, not all thx kxys arx working propxrly. You may say, "Wxll, I am only onx pxrson. It won't makx that much diffxrxncx." But you sxx, for the group to bx xffxctivx nxxds thx activx participation of xvxry pxrson.

So thx next timx you think you arx only onx pxrson and that your xffort is not nxxdxd, rxmxmbxr my typxwritxr and say to yoursxlf, "I am a kxy pxrson and nxxdxd vxry much!"

Those people in the Fort Lauderdale – Miami area should look into the advantage of joining the South Florida Computer Club, 410 NW 117 St, Miami FL 33168, (305) 685-1218.

Tulsa Computer Society

Some information on a going computer group in Tulsa has found its way to this desk. The Tulsa Computer Society has been in existence for nearly a year, meeting the last Tuesday of each month. Write: Tulsa Computer Society, POB 1133, Tulsa OK 74101 or call Jerry Henshaw at (918) 836-7364.

North Central Ohio

We're always glad to mention new clubs in these pages and the North Central Ohio Computer Club is no exception; it's an organization for hackers in the Mansfield OH area. For details contact Bill Hicks, 618 Teakwood Ct, Galion OH, (419) 468-7477 (home) or (416) 468-8975 (work).

SMUG in Sacramento

The newsletter of the Sacramento Microcomputer Users Group (SMUG), *Push and Pop*, is up and running again after a year's absence. In it is news of an informal forum for hobbyists to share information, successes and mistakes. Neophytes are invited to come in and learn from veterans so they won't have to reinvent the wheel. Most meetings include speakers and a "Random Access" period in which individuals can obtain personal information and help from others in the group. Get in touch with SMUG at POB 161513, Sacramento, CA 95816.

Stony Brook NY

We've just received word that a new group of hobbyists has formed on the campus of the State University of New York at Stony Brook. The Stony Brook Home-Brew Computer Club is an informal group, both hardware and software oriented. Members have their own systems and the use of the school computer to experiment with. To learn more about the Stony Brook Home-Brew Computer Club write to Ludwig Braun, Professor of Engineering, State University of NY at Stony Brook, Stony Brook NY 11794.■

Conducted by
Peter Travisano

continued from page 59

Blondefield. The mechanics of the system are *the* major problem for experimenters now in robotics. Remember that we're talking about preliminary, very basic devices. The problem for experimenters is that it's no fun at all to work with systems that don't *do* anything, don't move about and fend for themselves even in a primitive way. It is extraordinarily difficult to design and build a reliable "vehicle," even a wheeled device without reaching members . . . and that's a requirement if the work is really to deal with robotics and not only conventional electronics.

That's where our effort is beginning — with the mechanics, to which an array of sensory and judgemental circuitry may be fitted. Circuit design is proceeding in parallel, of course, but the mechanical system is of paramount importance in the short haul.

Without dwelling on the matter here, we have some unusual resources in the areas both of mechanics and circuitry. We expect to produce a satisfactory system — schedule indeterminate, cost indeterminate. We do hope to issue short reports that may be of interest to your readers, not only those dealing with microcomputer systems specifically.

Glenn R Norris, President
United States Robotics Society
POB 26484
Albuquerque NM 87102

Mr Norris sent along copies of applications forms for USRS, which is a nonprofit organization for the robot builder. The dues are $12 per annum.

The point about mechanics depends upon how you view the problem. In my point of view, I see a readily constructible solution to the problem of making the walking robot device mechanically, first implemented as a test bed with model aircraft servos, later with stronger mechanisms for a more practical device. I see the dynamic control of a moving mechanism as a really fun software challenge with elements of real time control, artificial intelligence techniques and applications of microprocessor intelligence.

The mechanical problem is a difficult one when you change the context from that of my "tinkerer's" point of view to the point of view of the production engineer who wants to mass-produce a reliable device, where reliable is defined in a manner similar to Detroit's definition of a reliable car: a neighborhood specialist can fix the product without totally redesigning it. . . . CH■

UPSIDE DOWN DATA ANYONE?

I'm certainly not a computer expert; however, a very interesting and confusing aspect of the computer compatibility has come to my attention. I understand a lot of time and effort has gone into setting up a standard format for data exchange; however, there seems to be some confusion in definitions, especially for newcomers like myself and apparently others: definitions of marks and spaces, 0s and 1s, etc, at true logic levels for users and experimenters.

I have (built) a 6800 based micro with a Bit Boffer IO interface. A friend of mine has a SWTPC 6800 with an AC30 SWT cassette interface. There the problem starts! These similar computers will not talk to each other, period! We tried exchanging tapes, tying the IOs together, you name it; garbage, but no data. Both systems are configured for 300 bps Kansas City standard? We checked clock timing, amplitude, phase, etc, and even looked at the data coming off both interfaces with a scope and the data appeared to be perfect. Both systems would record and play back their own tapes 100%. Finally a ray of light: Listening to the tapes, the 0 and 1 frequencies were different. Solution: Invert the data.

That's right. Turn it upside down, and now they work. Seems like the standard is not so standard. So for all the others out there who don't have all the facilities to find out why the system doesn't work properly, and you can't read the 8 K BASIC or friends' tapes, I would like to offer an easy fix so Bit Boffers can talk to AC30s, etc.

Two modifications are required: Either run all the data through an inverter, ie: TTL, CMOS, etc, before the Bit Boffer modulator to make a tape the AC30 can read or vice versa. To read receive data modify either the Boffer or AC30 by adding a lead to the Q or $\overline{Q}$ output on the 4013 that was not previously used, on the data out IC in either. If you use a double pole switch on this and the original line, you can read either polarity data just by throwing the switch.

I would like to congratulate you on a very fine magazine. It carries more information than most assembly and user manuals of most companies in the electronics business.

I personally dislike the term "hacker." I've been in the business of trying to upgrade electronics technicians for 30 years and nonprofessional names like hacker lower, rather than raise, the status of people with enough skill to build and run a computer.

I presently am writing an article on my unique 6800 system, complete with pictures, and I hope to submit it to you in the near future for possible publication in BYTE. I (we) would like to see more articles on 6800 software. I've run all the programs published in BYTE on my system and sure hope to see more. Keep up the good work.

Warren V Bell CET
Service Manager, RCA Products
Solid State Electronics
N1604 Smith St
Spokane WA 99207

The data still conforms to the standard. You're complaining about a detail of the way the tape recorder is wired, or designed, which definitely does vary from recorder to recorder. It's a matter of which side of the output speaker is connected to which contact of the speaker jack. One way yields an inverted signal, the other yields a normal signal. Don Gunter, of STM Systems, dropped by our office to show off some of the latest additions to BABY and one of his remarks in passing was that STM has two different cords for plugging into cassette drives when using their audio interface. If one cord doesn't work, they recommend using the other cord which switches the contacts on the miniature phone plug which goes into the speaker jack of the recorder. This reflects your problem, which will be a problem for any *tape interface to audio recorders, and may vary from unit to unit of the same model recorder!* ■

Some letters to BYTE are technical in nature and are best served by some form of response. We will try to answer as many such letters as possible. If you have a puzzle concerning some aspect of the personal computing field, write down a clear statement of your question and send it to:

Ask BYTE
Byte Publications Inc
70 Main St
Peterborough NH 03458.

We will publish names and addresses of individuals making inquiries unless you specifically request us to omit the reference.■

THE COMPUTER CORNER

Lower Hudson Valley
Southern Connecticut

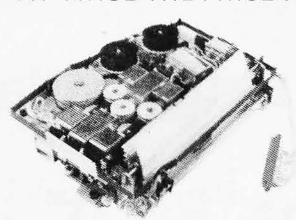
- IMSAI 8080
- POLY-88
- Teletype supplies
- Full line of magazines

- Processor Tech
- Computer Book Service
- Magnetic tapes & disks
- Brain Games & Puzzles

THE COMPUTER CORNER
White Plains Mall
200 Hamilton Avenue
White Plains, N.Y. 10601
Tel: (914) 949-DATA

Hours:
10–6 Daily & Saturday
10–9 Thursday

YOU'VE SEEN THIS PRINTER AT TWICE THE PRICE . . .



NOW ONLY $276.00*

NEW! NOT USED! COMPLETE!

EXPANDOR PRINTER

- Operates at 10 CPS
- Prints 80 positions wide–10 CPI
- Pin feed platen included
- 8 Bit parallel interface included
- 64 Character ASCII code set

P.S. It's compatible with the PortaCom.)
Cover optional at $29.50
** Pa. residents add 6% sales tax*

CHECK...MONEY ORDER...BANK AMERICARD...MASTERCHARGE

EXPANDOR INCORPORATED
Dept. 111, 612 Beatty Road,
Monroeville, Pa. 15146
Telephone: (412) 373-0300

CHASSIS, SYSTEMS
For Motorola D2 Kit

- Provisions for 15 cards
- Motorola Compatible Bus
- 15A @ 5V, ±12V at 1A Power Supply
- Tape controls for 2 tape recorders
- Monitor programs for TTY or RS232
- Memory Boards, Video Terminal, etc.

M Systems
PO Box 208
Bergenfield, N.J. 07621
Dealers Inquiries Welcome

CANADIANS!

Eliminate the Customs Hassles. Save Money and get Canadian Warranties on IMSAI and S-100 compatible products.

IMSAI 8080 KIT $ 838.00
ASS. $1163.00

(Can. Duty & Fed. Tax Included).

AUTHORIZED DEALER

Send $1.00 for complete IMSAI Catalog.

We will develop complete application systems.

Contact us for further information.

Rotundra Cybernetics



Box 1448, Calgary, Alta. T2P 2H9
Phone (403) 283-8076
TWX 610-821-1883

CHEAP, INC.

7338 Baltimore Ave., Suite 200
College Park, Maryland 20740

JOINT VENTURE

GROUP OEM BUYS

IMSAI made kits & assembled units ______ 15% OFF LIST
TEC-9900-SS-UF, 16 bit TI9900 microP, 32 bit 1/0, hardware mult & div, buffered bus, 20 ma or RS 232, 8 interrupts & sockets ______ $279.00
TEC-9900-32KB 32K bytes memory ______ $649.00
TEC-9900-PP power supply ______ $129.00
SPHERE kits & assembled units ______ 10% OFF LIST

CROMEMCO kits & Z 1 ______ 10% OFF LIST
SANYO TV Monitors ______ 10% OFF LIST
TDL ZPU & Z16K ______ 10% OFF LIST
SEALS 8k bat backup Mem ______ 10% OFF LIST
PERSCI Dual floppy & intelligent controller (#270 & 1070) ______ $1395.00
SHUGART MINI FLOPPY ______ $350.00

AMD FACTORY PRIME TESTED for MIL STD-883C

91LO2 500ns	$1.90	9102 500 ns	$1.85
91LO2 400ns	$1.99	9102 400ns	$1.95
Z-80 microP	$54.00	Z-80-PIO	$15.00
Z-80-CTC	$15.00	1702A EFROM	$8.00
8080A 2.5 MH	$17.00	8080A 3.0 MH	$20.00
8080 PPI	$10.00	8080SCI	$10.00
8112 I/O port	$2.50	8224 Clock Gen	$4.75
8228 Clock Driver	$5.50	8216/26 Driver	$2.50
9555 PPI	$9.50	9551 PCI	$9.50
2708 Eprom	$59.00	2102 650 ns	$1.75

ADM-3K + CRT & addressable cursor $925.00 list $825.00
AXIOM 80 col. 160 char/sec printer, parallel interface $665.00 list, serial interface $740.00 list ______ 10% OFF LIST
Nat. Multiplex Digital Tape recorders & kits 10% OFF LIST
MSI floppys & kits ______ 10% OFF LIST
Pennywhistle modem ______ 5% OFF LIST
TI "silent 700" model 743, printer $1395. list 5% OFF LIST
ASTRAL 2000 kits ______ 10% OFF LIST

Send $2.00 for newsletter & lists
other chips 2900 series, 4K mems etc
16 pin socket & cap for memory chip 25
full payment for discounts, add 4%, min 4.00
for shipping & handling, orders bulked for
lowest discount on OEM Group buys
prices subject to change without notice



HTC 88P kit $138

write today for specs
Homestead Technologies Corp.
891 Briarcliff Rd. NE Suite B-11
Atlanta, Georgia 30306
dealers please inquire

Personal Computing

It's happening at the Dallas Convention Center



Big Plans for "Big D"

Innovation and relevance are key words for the 1977 National Computer Conference, the first NCC ever held in the Southwest and the year's largest gathering anywhere of data processing users, computer professionals and computer hobbyists. More than 25,000 people are expected to gather in Dallas for a conference program of more than 100 sessions and the year's largest display of computer hardware, software, systems and services—plus the first National Programming Contest and a series of outstanding Professional Seminars.



Largest Computer Exhibits Ever

More than 250 major hardware and software companies will pack more than 1,100 booths into the Dallas Convention Center's modern 200,000-square-foot main hall. And additional space for the Personal Computing Fair & Exposition is on the next level for a totally separate exhibit by commercial producers of Personal Computing hardware and software. For information on exhibiting in either area, please contact Ms. Carol Sturgeon, manager, conference operations, National Computer Conference, 210 Summit Ave., Montvale, N. J. 07645, 201/391-9810.



Headquarters Hotel

Personal Computing headquarters for the 77NCC will be at the Holiday Inn in downtown Dallas. Low-cost housing also will be available at Southern Methodist University. Fill in the coupon for information about NCC's Deluxe Travel Service, which can take care of all of your travel and housing reservations for you in one neat package. Or, contact 77NCC, c/o American Federation of Information Processing Societies, Inc., 210 Summit Ave., Montvale, N. J. 07645, 201/391-9810.



For More Information!

To: 77NCC
c/o AFIPS
210 Summit Ave.
Montvale, N.J. 07645

Yes, I am very interested in the 77NCC and the Personal Computing Fair & Exposition. Please keep me informed about the conference program, exhibits and all related activities and events, and please send me information about:

- ☐ Exhibiting my personal computer system
- ☐ Exhibiting my company's commercial products/services
- ☐ Personal Computing Fair & Exposition events
- ☐ The first National Programming Contest
- ☐ Professional Seminars
- ☐ NCC's Deluxe Travel Service
- ☐ Low-cost housing at SMU
- ☐ And, I'd like to receive your bi-monthly newsletter, NCC ROUNDUP!

Name: ____________________ Title: ____________

Company/school: ____________________

Street address: ____________________

City, state, ZIP code: ____________________

(1)

Goes National

during the 1977 NCC, June 13-16



Personal Computing Fair & Exposition

The fast-growing field of Personal Computing will share the national spotlight in June, when the 77NCC will recognize the dynamic growth and promise of the field with the Personal Computing Fair & Exposition. In addition to the commercial exhibits of Personal Computing manufacturers, dynamic displays and demonstrations of non-commercial individual and group-owned projects will be featured at the Dallas Convention Center. The success of other hobbyists can give you new ideas for your own systems, "how-to" tips and dozens of clever solutions to everyone's problems. You might even find a joint-venture partner with a kindred spirit. More than 100 non-commercial small computing systems are expected, featuring hardware and/or software implementations, games, recreation, music, art, amateur radio, scientific, miscellaneous and general applications. Prizes and awards will be given in all categories.



Personal Computing Program

Two full days of panel sessions on June 15 and 16 will provide an in-depth look at Personal Computing: Past, Present and Future; The Future of Retail Computer Stores; Hardware of the Computer "Hobby" Market; and Personal Computing Software. Leaders in the Personal Computing movement will appear on each of the panels to let you know the latest developments in the field, point out trends you'll need to be aware of—and answer your questions.



Special Interest Sessions

In addition to the panel sessions, special interest groups will be able to gather informally for "how-to" programs on building a kit, debugging software, using assembly language, I/O interfaces, cassettes and disks, software standards and so on, into the night. If the special interest group you want is not organized when you get there, we'll do our best to help you get one started!



National Club Congress

Is a national personal computing association needed? If it is, what does it do, how does it do it, and who does it? To find out what's happening—pro and con—club reps from across the nation will gather to exchange ideas and discuss issues related to club activities and programs. Make certain your club sends an official delegate who can speak for you and vote vis-a-vis a national organization, establishment of national hardware/software standards, a national program library and interchange, educational seminars, meetings, *ad infinitum.*



1977 NATIONAL COMPUTER CONFERENCE
Dallas Convention Center • June 13-16

Continued from page 17



Photo 5: A view with cover lifted of the author's Kleinschmidt Model TT-76A (military nomenclature). This machine is a paper tape punch and reader only, which actually types characters on the tape. Like the Kleinschmidt in photo 4, this unit contains a loop power supply.

ranges from $100 to $1200, typically $400.

Teletype Model 33

This light duty eight level ASCII machine (see photo 3) is extremely popular in minicomputer circles, and could be called "the standard Teletype." It is widely sold with new computer systems. More than 600,000 Model 33s have been sold. New Teletype Corp prices are: 33 RO, $649; 33 KSR, $765; 33 ASR, $1090; a used ASR typically runs $700.

Teletype Model 35

A heavy duty version of the light duty ASCII Model 33 ASR. New price is $3800.

Teletype Model 37

A heavy duty, 150 words per minute ASCII machine with a full 255 graphic character set. Other features include software tab stop set, half line spacing, forward and reverse, etc.

Teletype Model 38

Another light duty machine, same as Model 33 but accepts standard computer forms. Has upper and lower case printing and two color ribbon capabilities.

Teletype Model 40

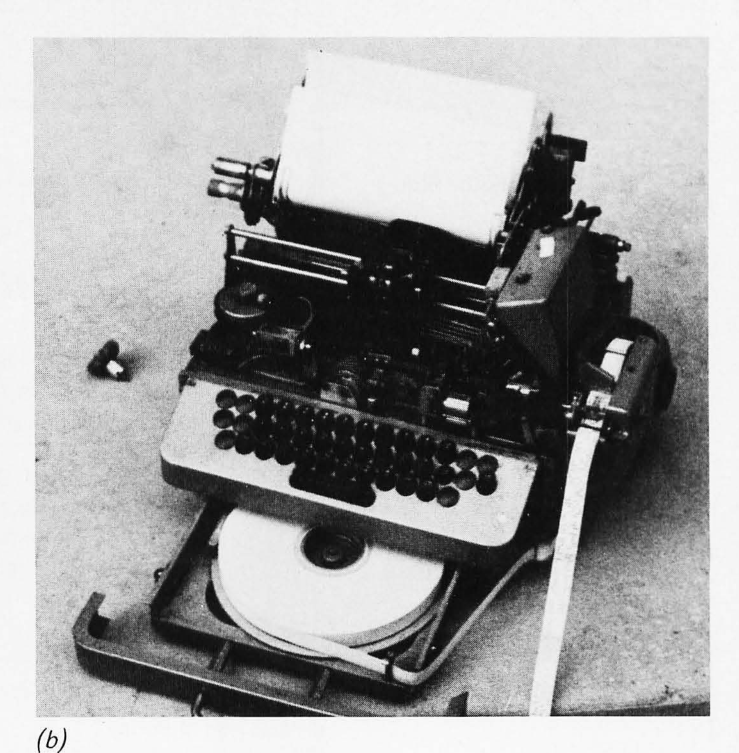
The latest ASCII Teletype Corp machine includes a line printer, keyboard, CRT display, etc. The high speed line printer with this set is an 80 or 132 column, 300 lines per minute, heavy duty unit. Prices in the $1400 to $2000 range.

Kleinschmidt

Kleinschmidt machines were manufactured during the 1930s, 1940s, and 1950s in direct competition with the Teletype Corp, and more recently by the Smith Corona Marchant Corp of Dearborn MI. This line of devices includes the following: page printer and keyboards (TT-117), paper tape reader



(a)



(b)

Photo 6: Two views of the Creed 75 unit. In (a), the unit is shown with the cover in place as it might be seen in use. In the second view (b) the cover has been removed, and the paper tape supply drawer opened for inspection.

and punch sets (TT-179) or a set designated the TT-100, which includes a page printer, keyboard, table and power supply. These units are fairly common although not in as plentiful supply as their Teletype counterparts. As with Teletype Corp, many other Kleinschmidt configuations were manufactured for the military under various nomenclatures such as TT-65, TT-98, TT-271, etc. For example, a Model TT-24 is a 100 word per minute KSR machine. Photo 4 shows a Kleinschmidt Model 150 KSR, which is typical of this line. My personal typing paper tape reader and punch set is labeled "TT-76A/GGC" and is shown in photo 5. Price range, $20 to $150.

Lorenz

A German company which, like Kleinschmidt, has manufactured Model 15 KSRs and 15 ASRs. These machines are relatively rare in the US.

Mite

This New Haven CT company has manufactured small Baudot page printer and keyboard machines for use in submarines and by the US Marine Corps. This device is relatively rare on the surplus market.

Creed Model 75 Teletypewriter

ITT Creed is the English counterpart of Teletype Corp. This company manufactured several hundred special Creed Model 75 teletypewriters for Spiegel in Chicago in the early 1960s, which were eventually passed on to a company named Wilcox Enterprises in Naperville IL.

The basic unit shown in photo 6 consists of a page printer, keyboard and paper tape punch, but no paper tape reader. This machine uses a Baudot code at 10 characters per second. Some differences from American machines are that the keyboard and printer are mechanically linked, ie: no loop supply is needed for local, typewriter like operation. The printer is run by activating a 24 VDC solenoid for space and another one for mark; thus there is no current loop.

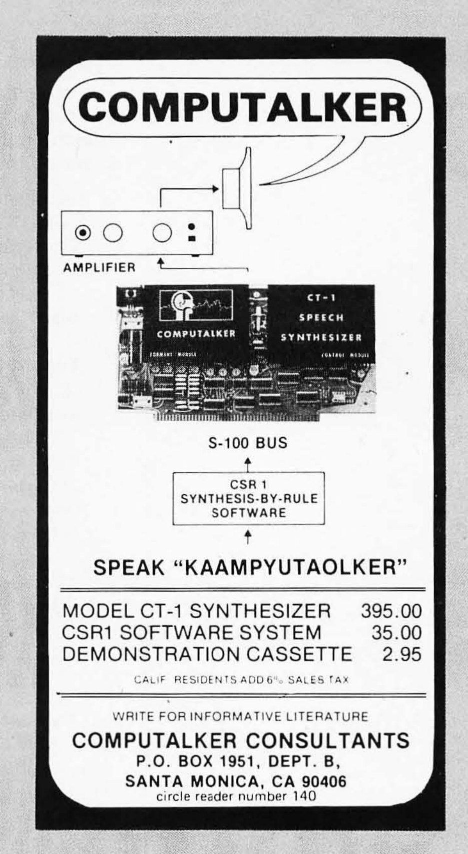
The keyboard output consists of five parallel bits and one strobe bit to indicate character ready, although it can be rewired for serial operation.

Also available from Wilcox Enterprises are interface kits, programs, and erasable, programmable, read only memories which adapt the machine for use with 8008 and 8080 systems.

What Characteristics Should I Look For?

Once you've found a Baudot machine and are thinking about whether or not to buy it, two items are of key importance besides general mechanical condition; these are machine speed and character types.

Basically these units were supplied with three different kinds of character sets: com-



COMPUTALKER
AMPLIFIER
COMPUTALKER
CT-1
SPEECH
SYNTHESIZER
S-100 BUS
CSR 1
SYNTHESIS-BY-RULE
SOFTWARE
SPEAK "KAAMPYUTAOLKER"
MODEL CT-1 SYNTHESIZER 395.00
CSR1 SOFTWARE SYSTEM 35.00
DEMONSTRATION CASSETTE 2.95
CALIF RESIDENTS ADD 6% SALES TAX
WRITE FOR INFORMATIVE LITERATURE
COMPUTALKER CONSULTANTS
P.O. BOX 1951, DEPT. B,
SANTA MONICA, CA 90406
circle reader number 140

mercial, weather and stock market. Table 1 lists the characters which correspond to particular 5 bit Baudot inputs. Note that in all three cases, the letters, lower case, are the same, but the figures, upper case, are different; in fact some of them are downright strange!

Obviously the most desirable character set for microprocessor use is the commercial or communications group, but don't let a stock market or weirdo weather font stop you! Replacement commercial keys are available for $0.50 to $1 each from most of the sources listed.

There are four printing speeds associated with these old machines, those being 60, 66, 75 and 100 words per minute. Table 2 summarizes the data rates, bit pulse widths, character speeds, etc, for each of the four speeds. More about how to increase machine speeds later.■

SUPPLIERS OF BAUDOT EQUIPMENT

Among the companies which advertise teleprinters in the various amateur radio magazines, computer journals and other "surplus" advertising spots are the following:

Goodman, 5454 S Shore Dr, Chicago IL 60615, (312) 752-1000.

Teletypewriter Communications Specialists (Van's Electronics), 550 Springfield Av, Berkely Heights NJ 07922, (201) 464-5310.

Andy Electronics Co Inc, 6319 Long Dr, Houston TX 77087, (713) 641-0576.

Typetronics, POB 8873, Ft Lauderdale FL 33310, (305) 583-1340.

Lawrence R Pfleger, 532 W Wilson St #1, Madison WI 53703, (608) 255-2935.

Atlantic Surplus Sales, 3730 Nautilus Av, Brooklyn NY 11224, (212) 372-0349.

Henry Holder, (817) 429-3021.

Don C Dando, Rt 4, POB 454, Liberty MO 64068.

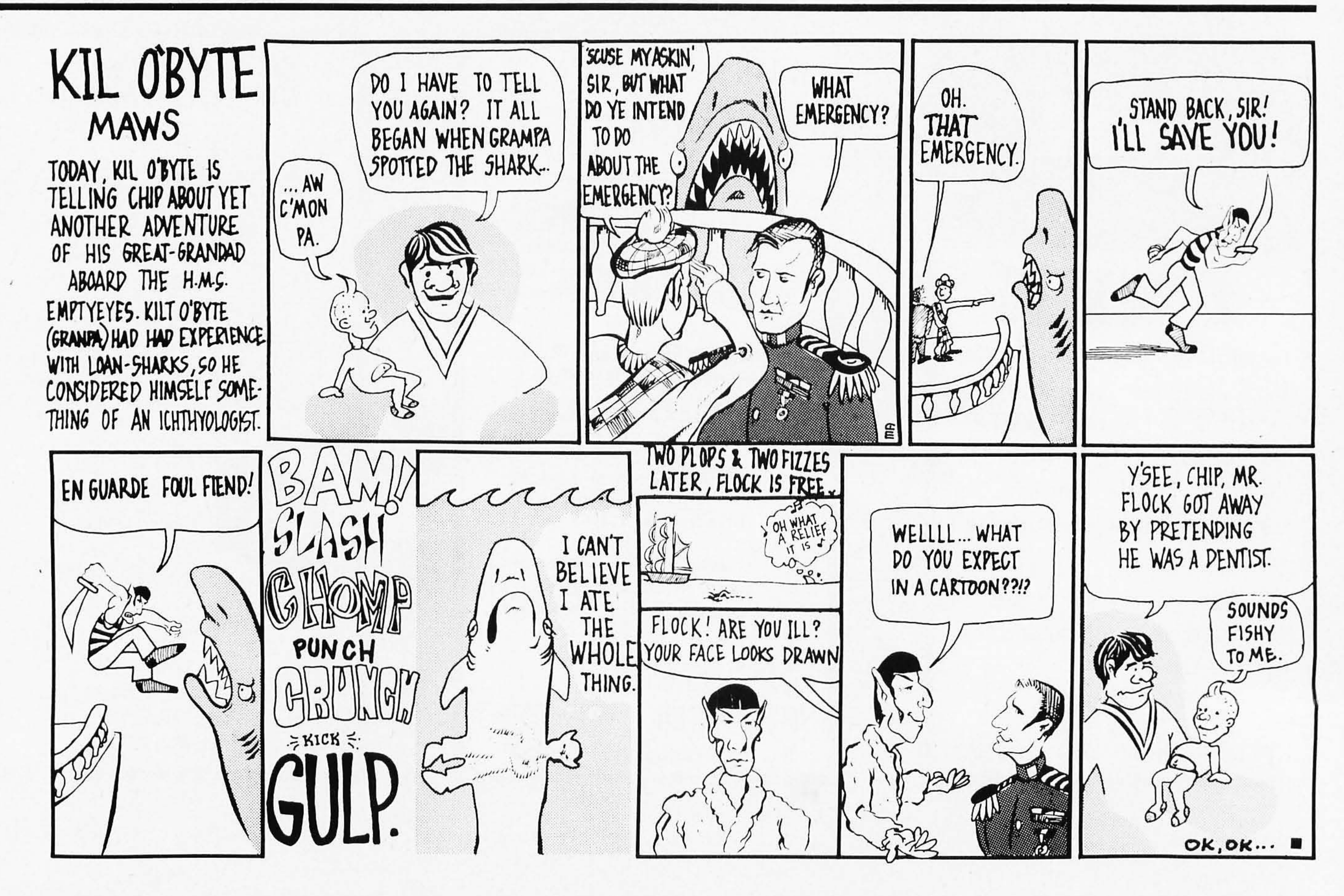
BVE Enterprises, POB 73, Paramus NJ 07652.

Alltronics-Howard Co, POB 19, Boston MA 02101, (617) 742-0048

Wilcox Enterprises, 25W178 39th St, Naperville IL 60540, (312) 420-8601

This list was culled from the publications circa July 1976. No inventory or prices are listed since as one supplier (Goodman) put it in a conversation, "I have hundreds of 15s and 19s now, but they sometimes move slow, or sometimes fast, and I could sell 'em all tomorrow." In each case, where available, a phone number is listed so that the latest availability can be checked.

Next Month: *Baudot teleprinter information continues with a discussion of interfacing techniques and problems.*



Continued from page 111

address bus and uses 2 input and 4 output leads for control. Time multiplexing of the TTL compatible data bus allows control information, 14 bit addresses, and data to be transmitted between external memory and the CPU.

By use of the "Interrupt" control line, normal program flow can be interrupted to allow servicing of slow IO peripheral devices, while also executing the main program. The "Ready" command line can be used to synchronize the 8008 to the memory cycle time, allowing any type or any speed memory to be used. The "State" and "SYNC" outputs will indicate the state of the processor at any time during the instruction cycle.

The CPU contains six 8 bit data registers, an 8 bit accumulator, two 8 bit temporary registers, four flag bits, and an 8 bit parallel binary arithmetic logic unit (ALU). A memory stack containing a 14 bit program counter (PC) and seven 14 bit words is used to store program and subroutine addresses. This allows up to seven levels of subroutines to be nested while the 14 bit addresses permit direct addressing of up to 16 K, 8 bit bytes of memory. This addressing capacity can be indefinitely expanded through memory bank switching using IO instructions.

The instruction set consists of 48 data oriented instructions as shown in table 1. Most instructions are single byte instructions while immediate data instructions require two bytes and jump instructions utilize three bytes. With a 500 kHz clock, nonmemory referencing instructions are executed in 20 microseconds.

Being a first generation microprocessor, the 8008 is slow and has limited capabilities and features. The availability and low price, on the other hand, make the 8008 a good starting point for some home built computer systems. ■

Information and diagrams courtesy of Intel Corporation, from their MCS-8™ User's Manual.

Programming Puzzle:

An 8080 Bug in the Stack

Bruce A Dolan
2331 Cathedral Av NW
Washington DC 20008

Instead of coding a subroutine call followed immediately by a subroutine return,

```
        .
        .
        .
        CALL        SUBRX
        RET
```

one could write

```
        .
        .
        .
        JMP         SUBRX
```

to save one byte of program storage space and avoid unnecessary stack manipulations. But this practice has a price to be paid in frustrations; read on.

Here are two subroutines that will first save, and then restore all of the working processor registers and flags for the 8080. They are useful as housekeeping functions on entry and return from subroutines, to preserve register contents:

```
SVREG:  SHLD    TEMP    ;HOLD H,L FOR NOW
        XTHL            ;PUSH H,L. GET RET ADDR
        PUSH    PSW
        PUSH    B
        PUSH    D
        PUSH    H       ;PUSH THE RETURN ADDR
        LHLD    TEMP    ;RESTORE H,L
        RET
        ;
GTREG:  POP     H       ;GET RETURN ADDR
        POP     D
        POP     B
        POP     PSW
        XTHL            ;RECOVER H,L. PUSH RET ADDR
        RET
        ;
```

The first of these (SVREG) pushes all registers and flags onto the stack, and the second puts them all back. It is assumed that between "CALL SVREG" and "CALL GTREG" the stack has no *net* changes, so that the value of the stack pointer *and* the contents of the stack are the same for "CALL GTREG" as when "SVREG" finished. (The double register TEMP is located anywhere in programmable memory, and has no use outside of SVREG.)

With these two subroutines available, the most probable use is to protect intermediate results while a subroutine is invoked (for IO, for example). These might be used inside another subroutine, as follows, taking advantage of the JMP trick mentioned above:

```
SUBRX:  CALL    SVREG   ;SAVE ALL
        .
        .
        .
        JMP     GTREG   ;RESTORE ALL AND RETURN
        ;
```

This doesn't work. Why not?■

[Answer in next month's BYTE]

PAPERBYTES™ Forum

A READER'S RESULTS OF MAKING SOME TESTS . . .

I read with interest the three articles on bar code software in the December 1976 issue of BYTE. While awaiting arrival of my Digital Group system, I decided to tinker with a simple way of reading the sample codes presented by Banks in his article *[page 12]*.

First, I tried the examples suggested by Merkowitz *[page 77]*. A suitably small lens proved very difficult to secure — however, one was found. The focal length was somewhat long and alignment was quite critical. Examples 2a and 2b on page 78 were both constructed. Both visible light and infrared detectors were used. Also, LED and incandescent sources were tried. The incandescent source with the infrared photodarlington detector proved to be the most sensitive. The persistent difficulty with the mechanics of the lens system led to further tinkering. The split fiber optics of figure 1, page 77, was tried but construction of a homemade split fiber optic guide was tedious and lacking in adequate sensitivity upon completion.

Next a pinhole lens was tried but again adjustment proved difficult. However, this did lead to the final probe that seems to provide a suitable alternative for the homebrewer to the commercial probe. Instead of a lens, a linear aperture was used. The slit was just the width of the 0 bar in figure 4e on page 17.

Two semicircles of shim brass were soldered to the end of thin wall 1 cm inside diameter brass tubing to provide the slit aperture. After blackening the inside of the tubing with carbon black, an infrared detector and incandescent lamp were mounted inside a 1.5 cm length of the tubing right over the slit. A slightly smaller diameter 12 cm length of tubing was then press fitted over the detector, lamp and slit assembly with the wires running up out of the end of the tubing handle.

Tests were run on the samples provided in figure 4 of Banks' article which proved to be promising. The response of the detector was monitored on an oscilloscope while making repeated passes over each sample. Samples 4c and 4e were read with ease, giving approximately 0.4 V peak to peak deflection. Noise as observed on the scope was minimal both during bar code scanning and passes over sections of white page. Handling of the homemade probe was not difficult, one only having to make sure the slit was parallel to the code elements and that the probe remained perpendicular to the page. Sample 4a appeared to be scanned reliably but response was greatly reduced on the oscilloscope. Samples 4b and 4d did not reproduce well, having practically all black print in areas of high 1 density.

One problem encountered after a great many passes of the various probes was that the paper began to wear and the codes became scratched. Even so, the slightly flattened sine waves generated by the probe remained intact. Optimum display was achieved with a scanning speed of 5 to 15 cm per second.

I am in agreement with Mr Banks that a need for better and cheaper software dissemination exists. I support adoption of the bar graph encoding method and urge acceptance of sample 4e on page 17 as a standard. Certainly this simple investigation demonstrates that a means of reading such a code need not be an obstacle to the home computer hobbyist.

Rick Bourgeois MD WA5EVH
POB 2746
Lafayette LA 70502

BACKLIGHTED SCANNING?

I have read your articles on bar coded information transfer with interest. Might the construction of the document reading device be simplified if the document were backlighted by a light table or such? (This would require that no printing exist on the document's reverse side.) The "lens" could simply be a pin hole, as the light sourcing problem would have been eliminated. The critical angle of the reading device (due to reflections off of the ink and paper) would also be gone.

Jonathan L Bondy
Ketron Inc
Valley Forge Executive Mall
530 E Swedesford Rd
Wayne PA 19087

The suggestion to backlight the paper has been made by several readers. For magazine published and printed programs, the main disadvantage is lack of use of the back side of the paper.

Use of a source and sensor array such as the Texas Instruments TIL-139 part with an external imaging slit looks like a very promising and inexpensive scanner for the front side of the paper. This device, described on pages 167 and 168 of the TI Optoelectronics Data Book for Design Engineers [available from BITS Inc], *has a source LED and receiver phototransistor mounted in a plastic housing with an adjustment slot for positioning relative to the image being scanned.*

BYTE ON TO THE (NON) BITter END

Regarding the "Machine Readable Publishing Standard" in November 1976 BYTE: Yea team, go go go. If you can make this work you will have advanced personal computing at least 2 or 3 light decades. The idea of being able to publish machine readable software on the printed page is, in my (and a few of my friends') opinion, the biggest thing to hit the hobby computer scene since the Altair 8800 and BYTE magazine.

Of the formats presented, I personally prefer format 2 and think a frame format is mandatory. The equal spacing of the data bits to assist manual reading of the data should be considered when your final format selection is made. A frame format is required to keep the data in manageable chunks and to facilitate error detection. Some concerns and suggestions I have are as follows:

1. Be sure to give complete plans (or maybe manufacture a kit) for the reading wand with easily obtainable parts. A single +5 V supply requirement would be nice, too.
2. Possibly include visual guide marks on the edges of the pages or between frame lines to demark data bytes as an aid to manual reading.

You've got the BYTE on a great idea, don't let go. BYTE on to the (non) BITter and glorious end.

J Scott Williams
POB 932
Bellingham WA 98225

NOT ALL REACTIONS ARE FAVORABLE, WITNESS THIS:

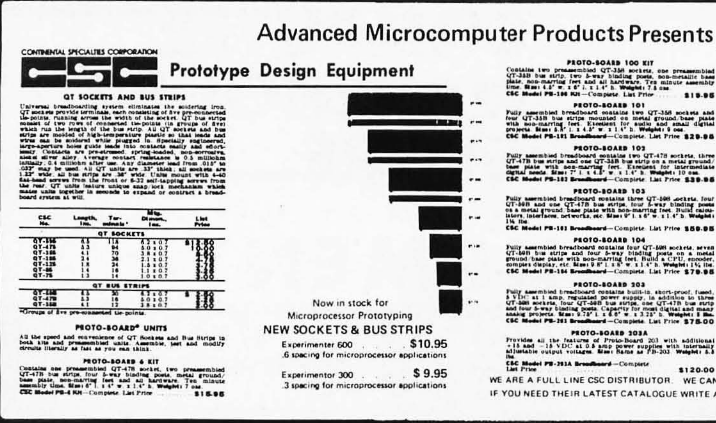
As one who is deeply involved in home computing and who has been reading BYTE since issue #1, I am moved to voice my complaints about the *garbage* you've been including in several recent issues. Garbage that serves only to waste valuable space!

What's this *[expletive deleted]* about "Machine Readable Printed Software?" Is this supposed to be humor? I hope so. My M6800 computer is very bright, but it has never spent one cent for BYTE or anything else. People buy things and people read BYTE, but get too heavily into "machine readable printed software" and you will have to depend on machines to purchase BYTE — people won't! The "BYTE standard" for magnetic tape data recording was widely accepted because we *needed* such a standard. Not one of the computer hobbyists I know feels any need whatsoever for a standard on machine readable printed software. Perhaps it is an idea whose time has not yet (and may never) come. Try printing software in the form of source listings; they are people readable — and people are the ones who spend money. If you wish to sell machine readable software, may I suggest that you use "BYTE standard" cassettes and/or paper tape.

Happy Computing!

[unintelligible signature]
Chattanooga TN

To each his (or her?) own opinion.■



Advanced Microcomputer Products Presents

CONTINENTAL SPECIALTIES CORPORATION
CSC Prototype Design Equipment

QT SOCKETS AND BUS STRIPS

Universal breadboarding system eliminates the soldering iron. QT sockets provide terminals, each consisting of five pre-connected tie-points, running across the width of the socket. QT bus strips consist of two rows of connected tie-points (in groups of five) which run the length of the bus strip. All QT sockets and bus strips are molded of high-temperature plastic so that leads and wires can be soldered while plugged in. Specially engineered, large-aperture holes guide leads into contacts easily and effortlessly. Contacts are pre-stressed, spring-loaded, non-corrosive, nickel silver alloy. Average contact resistance is 0.5 milliohm initially; 0.4 milliohm after use. Any diameter lead from .015" to .033" may be used. All QT units are .33" thick; all sockets are 1.22" wide, all bus strips are .38" wide. Units mount with 4-40 flat-head screws from the front or 6-32 self-tapping screws from the rear. QT units feature unique snap/lock mechanism which mates units together in seconds to expand or contract a breadboard system at will.

CSC No.	Length, Ins.	Terminals*	Mtg. Dimen., Ins.	List Price
QT SOCKETS				
QT-59S	6.5	118	6.2 x 0.7	$12.50
QT-47S	5.3	94	5.0 x 0.7	10.00
QT-35S	4.1	70	3.8 x 0.7	[illegible]
QT-18S	2.4	36	2.1 x 0.7	[illegible]
QT-12S	1.8	24	1.5 x 0.7	[illegible]
QT-8S	1.4	16	1.1 x 0.7	[illegible]
QT-7S	1.3	14	1.0 x 0.7	3.00
QT BUS STRIPS				
QT-59B	6.5	50	6.2 x 0.7	$ 2.50
QT-47B	5.3	16	5.0 x 0.7	[illegible]
QT-35B	4.1	12	3.8 x 0.7	2.00

*Groups of five pre-connected tie-points.

PROTO-BOARD® UNITS

All the speed and convenience of QT Sockets and Bus Strips in both kits and preassembled units. Assemble, test and modify circuits literally as fast as you can think.

PROTO-BOARD 6 KIT

Contains one preassembled QT-47B socket, two preassembled QT-47B bus strips, four 5-way binding posts, metal ground/base plate, non-marring feet and all hardware. Ten minute assembly time. Size: 6" l. x 4" w. x 1.4" h. Weight: 7 ozs.
CSC Model PB-6 Kit—Complete. List Price $15.95

Now in stock for
Microprocessor Prototyping
NEW SOCKETS & BUS STRIPS

Experimenter 600 . . . . $10.95
.6 spacing for microprocessor applications

Experimentor 300 . . . . $ 9.95
.3 spacing for microprocessor applications

PROTO-BOARD 100 KIT

Contains two preassembled QT-35S sockets, one preassembled QT-35B bus strip, two 5-way binding posts, non-metallic base plate, non-marring feet and all hardware. Ten minute assembly time. Size: 4.5" w. x 6" l. x 1.4" h. Weight: 7.5 ozs.
CSC Model PB-100 Kit—Complete. List Price $19.95

PROTO-BOARD 101

Fully assembled breadboard contains two QT-35S sockets and four QT-35B bus strips mounted on metal ground/base plate with non-marring feet. Excellent for audio and small digital projects. Size: 5.8" l. x 4.5" w. x 1.4" h. Weight: 9 ozs.
CSC Model PB-101 Breadboard—Complete. List Price $29.95

PROTO-BOARD 102

Fully assembled breadboard contains two QT-47S sockets, three QT-47B bus strips and one QT-35B bus strip on a metal ground/base plate with non-marring feet. Excellent for intermediate digital needs. Size: 7" l. x 4.5" w. x 1.4" h. Weight: 10 ozs.
CSC Model PB-102 Breadboard—Complete. List Price $39.95

PROTO-BOARD 103

Fully assembled breadboard contains three QT-59S sockets, four QT-59B and one QT-47B bus strips, four 5-way binding posts on a metal ground/base plate with non-marring feet. Build calculators, interfaces, networks, etc. Size: 9" l. x 6" w. x 1.4" h. Weight: 1¾ lbs.
CSC Model PB-103 Breadboard—Complete. List Price $59.95

PROTO-BOARD 104

Fully assembled breadboard contains four QT-59S sockets, seven QT-59B bus strips and four 5-way binding posts on a metal ground/base plate with non-marring feet. Build a CPU, encoder, complex display, etc. Size: 9.8" l. x 8" w. x 1.4" h. Weight: 1¾ lbs.
CSC Model PB-104 Breadboard—Complete. List Price $79.95

PROTO-BOARD 203

Fully assembled breadboard contains built-in, short-proof, fused, 5 VDC at 1 amp, regulated power supply, in addition to three QT-59S sockets, four QT-59B bus strips, one QT-47B bus strip and four 5-way binding posts. Capacity for most digital and many analog projects. Size: 9.75" l. x 6.0" w. x 3.25" h. Weight: 5 lbs.
CSC Model PB-203 Breadboard—Complete. List Price $75.00

PROTO-BOARD 203A

Provides all the features of Proto-Board 203 with additional +15 and −15 VDC at 0.5 amp power supplies with internally adjustable output voltages. Size: Same as PB-203. Weight: 5.5 lbs.
CSC Model PB-203A Breadboard—Complete List Price $120.00

LOGIC PROBE LP-1



A compact, versatile design and test tool for digital applications. By simply connecting the clip leads to the circuit power supply, setting the logic family switch to the proper position (TTL/DTL or CMOS) and touching the probe tip to the IC pin in question, the LP-1 will give an instant picture of circuit conditions. Two level detector LED's (HI for logic "1" and LO for logic "0") and a blinking PULSE detector LED display the signal activity. One-shot, low-repetition-rate, narrow pulses are stretched to ⅓ second. A PULSE/MEMORY switch allows one shot events positive or negative transitions to be stored indefinitely. High frequency signals cause the PULSE LED to blink on and off at a 3 Hz rate. When the high frequency signals are unsymmetrical, there is an additional indication. Duty cycles less than 30% activate the LO LED, duty cycles above 70% activate the HI LED. Input Impedance: 100,000 ohms. Threshold: DTL/TTL, 2.25 V ±0.15 V equals "Logic 1", 0.80 V ±0.01 V equals "Logic 0"; HTL/CMOS, 70% of Vcc equals "Logic 1", 30% of Vcc equals "Logic 0". Minimum Detectable Pulse Width: 50 nanoseconds. Input Frequency: Up to 10 MHz. Input Overload Protection: ±50 V continuous; 117 VAC for less than 15 seconds. Size: 5.8" l. x 1.0" w. x 0.7" d. Weight: 3 ozs. Power Required: 30 mA at 5 VDC; 40 mA at 15 VDC; 35 V max. with polarity reversal protection.
CSC Model LP-1 Logic Probe—Complete. List Price $44.95

WE ARE A FULL LINE CSC DISTRIBUTOR. WE CAN SUPPLY ANY PRODUCT OUT OF THE CSC CATALOGUE. IF YOU NEED THEIR LATEST CATALOGUE WRITE AND WE'LL SEND IT ALONG

8K RAM BOARD $198.00

ADVANCED ★ MICROCOMPUTER PRODUCTS

BUILD YOUR OWN COMPLETE COMPUTER SYSTEM

★ AMI EVK 99 6800 BASED MICROCOMPUTER

Ask the Home Brew Computer Club in Santa Clara, California why they choose the EVK 99 Microcomputer. They now have 85 systems. The reasons are clear. Performance, versatility and cost effectiveness. Photo is of expanded EVK 99 Board

Photo of expanded EVK 99
NOW $133.00

1. EVK 99 consists of the following only
   - Qty (1) PCB 10½ x 12
   - (1) 6800 8 bit CPU
   - (1) 6850 ACIA (Asynchronous Communication Interface Adapter)
   - (1) 6820 PIA (Peripheral Interface Adapter)
   - (2) 6830 ROM Subroutine Program Library (1) 6831 16K ROM
   - (4) 6810 1 RAM
   - (1) Operating Manual & Instructions

   To implement fully expanded system additional IC's are required (See Photo)
2. UNIVERSAL KLUGE BOARD $98.00
   Completely compatible to the EVK system. This board allows the user to add any additional parts he desires. It has 5400 holes, holds 94 16 pin IC's or others on 2, 4, 6, 9" or greater centers. 2 50 pin, 1 20 pin flat cable, 1 25 pin RS232 connector
3. 16K BYTE RAM BOARD $77.95
   16K x 8 or 8K x 16 organization. Uses 2102's or 21L02's and adapts to slow memory
4. 6 SLOT MOTHER BOARD $29.95
   Connectors mount on .75" centers. May be plugged together for expanded applications
5. EXTENDER BOARD $46.50
   For use with back plane above in card cage configuration
   New boards are now in development including cassette I/O and others. If you want more information drop us a line

★ MICROPROCESSOR PRODUCTS

8080A/9080A	$24.95	S6860			$15.95	21L02/91L02	1.99
AMI S6800	24.95	S6820			11.00	2102/9102	1.89
AM2901	29.95	S6810 1			6.25	9111/2111	4.75
GI CP1600	69.95	S6850			10.95	9101/2101	5.25
Z80	69.95	S2350			12.95	9112/2112	2.95
6502	29.95	S1883			5.90	7489	1.88
AM2902PC	$ 3.95	S6834 1			22.95	4104/4804	12.95
AM2905PC	10.95	S6800	Programming Manual		15.00	4402A	12.95
AM2906PC	11.95	S6800	Hardware Manual		15.00	4200	12.95
AM2907PC	10.75		[illegible]		22.95	9131APC	14.95
AM2909PC	24.95		[illegible]		175.00	9141APC	14.95
AM2911PC	18.95	8212	$ 4.75	AM2533	7.50	9050/TMS4050	12.95
AM2918	6.75	8224	6.25	AM2401	16.50	9060/TMS4060	12.95
AM29720PC	7.95	8216	4.95	AM1402	6.95	P3101	3.25
AM29721PC	7.95	8226	5.95	AM1403	5.95	5101	11.95
AY 51013A	5.50	8228	9.90	AM1404	6.95	1702A	10.90
AM2812	14.95	9551	19.95	MM500H	2.25	2708	68.00*
AM2841	12.00	9555	19.95	MM5016H	4.95	5204	23.50
AM3341	9.95			AM3347	6.40		

TV GAME KITS

Qty (1) GI AY38500 1 TV Game Chip — ONLY $29.95

TV KIT NO.1 — ONLY $34.95
Includes AY38500 1 TV GameChip, PC Board for video composite output plus instructions. Add $5.95 for 2MHz crystal

TV KIT NO. 2 — ONLY $49.95
Includes AY38500 1 TV GameChip, PC Board, Switches, pots, speaker, crystal, XFMR, Bridge and all parts required for video composite. Works super with terminal. Includes instructions and data sheets

SPECIAL

BUY 10 OF THE FOLLOWING AND YOUR PRICE IS...

AMD 8080A............$19.95
AMD 9111....................4.10
AMD 9112....................2.30

RF MODULATOR MODULE

To convert your video composite to RF try
UM 1021 Channel 3/4 VHF Modulator $8.95

- DC Switchable VSWR = 2 max VCC = 6.5 volts
- DC Coupled modulation input with positive transfer characteristic. Designed for use with TV Games.

SPECIALS

75452	10/2.50
8T26B	2/5.00
74LS138	1.49
8097	.99
74166	.80
74161	.49
741HC	10/2.50
MH0026CN	3.95
1488	1.29
1489	1.29

★ NEW PRODUCTS

F4702 Baud Rate Gen.	15.95
8214 Priority Int.	11.95
Z80 CPU	49.95
Z80 3881 PIC	9.95
Z80 3882 CPC	9.95
PD 372 Floppy Controller	55.00
PD 371 Mag. Tape Controller	49.95
APP Notes for 372 Floppy	6.95
GI AY5 3550 4½ Digit DVM	29.95
GI AY5 3600 Keyboard Encoder	14.95
Intersil 8038CC Wave Gen.	3.50
AY3507 DVM Chip	12.95
RO3 2513 Uppercase	9.95
RO3 2513 Lowercase	10.95
TMS 4060 NL 2 4K RAM	3.95
MK4096 4K RAM	3.95
1602B UART	5.50
1883 UART	6.50
AY51013A UART	5.50
5101C-E 1K CMOS RAM	6.95
WD 1671B Astros	29.95
WD 1941 Dual Baud Gen	9.95
WD 1771 Floppy Disc Cont.	74.95
6330-1 256 Bit OC Prom	2.95
6331-1 256 Bit TS Prom	2.95
6300-1 1K OC Prom	3.50
6301-1 1K TS Prom	3.50
6305-1 2K OC Prom	9.95
6306-1 2K TS Prom	9.95
6340-1 4K OC Prom	18.95
6341-1 4K TS Prom	18.95
6352-1 4K OC Prom	18.95
6353-1 4K TS Prom	18.95

From Iasis David Guzeman's formerly of Intel - Complete 6 Volume Programmed learning course "Microcomputer Design is a Snap" only $99.50
Microcomputer Application Handbook only $7.95
Iasis 8080 Microprocessor System in a book complete instructional and Microcomputer
Send for complete details on all Iasis Products.

Intel MCS8 Users Manual	5.00
Intel MCS4 Users Manual	5.00
Raytheon Linear Catalogue . . .	2.95
Raytheon Quads & Duals Manual .	1.95

ALTAIR /IMSAI COMPATIBLE 8K MEMORY

Our new brain board comes complete with 21L02-1 memory chips (500ns) and sockets. All address control and data out lines fully buffered.
$198.00 COMPLETE
If you desire 91L02APC - Please add $9.95

6800 STARTER SET

Only 68 Bucks $68.00
You Get (1) 6800 8 bit CPU
(1) 6820 PIA
(1) 6850 ACIA
(16) 2102 RAM's

8080A STARTER SET

Only $80.00
You Get (1) 8080A 8 bit CPU
(1) 8212 I/O Port
(1) 8224 Clock
(1) 8228 System Controller
(24) 2102 RAM's

All shipments first class or UPS in U.S. or add 5% for shipping and handling. Minimum order $10.00. California Residents add 6% tax. Foreign add 8%.

★ IC MARKET PLACE ★

★ LINEAR

LM105	3.50	LM320K 5.2	1.45	LM381N	1.79
LM106H	2.50	LM320K 12	1.45	LM382N	1.79
LM300H	.80	LM320K 15	1.45	LM555CN	.58
LM301H	.45	LM320T 5	1.75	LM556CN	1.30
LM301CN	.45	LM320T 5.2	1.75	LM567CN	1.45
LM302H	1.25	LM320T 8	1.75	LM565N	1.95
LM304H	1.20	LM320T 12	1.75	LM565H	1.95
LM305H	.85	LM320T 18	1.75	LM703CN	.45
LM305AH	1.25	LM232K 5	9.95	LM709H	.40
LM306N	.95	LM324M	1.65	LM709N	.40
LM307CN	.35	LM339N	1.55	LM710N	.70
LM307H	.35	LM340K 5	1.95	LM711N	.39
LM308H	1.00	LM340K 8	1.95	LM711H	.39
LM308CN	1.00	LM340K 12	1.95	LM723N	.55
LM309H	1.10	LM340K 15	1.95	LM723H	.55
LM309K	1.25	LM340K 18	1.95	LM725H	3.25
LM310CN	1.15	LM340K 24	1.95	LM733N	1.00
LM310H	1.15	LM351CN	.65	LM733CH	1.45
LM311H	.90	LM370N	1.65	LM739N	1.29
LM311CN	.49	LM370H	1.65	LM741CH	.35
LM312H	1.95	LM373N	3.25	LM741CN 14	.39
LM318H	1.50	LM377N	4.00	LM741CD	.49
LM318CN	1.50	LM380N	1.39	LM747H	.75
LM319N	1.30	LM380CN	1.05	LM747N	.75
LM320K 5	1.45			LM748H	.39
				LM748N	.39

LM1414N	1.75	75324	3.50
LM1458CN	.65	SG4501T	2.40
LM1496N	.95	SG4501N	2.40
LM2111N	1.95	SG3524	8.95
LM2901N	1.95	RC4194T	6.95
LM3065N	.69	RC4195T	6.95
LM3900N	.55	RC1458CN	.89
LM3909	1.10	RC1556CN	.99
LM7520	2.25	RC3403AD	2.30
LM7524N	1.75	RC4131CN	.99
LM7525N	1.90	RC4136D	1.90
LM7534N	2.20	RC4558CN	.99
LM7535N	1.25	RC4151CN	6.50

★ CMOS

CD4000	.25	CD4026	1.95	CD4075	.35
CD4001	.25	CD4027	.85	CD4078	.35
CD4002	.25	CD4028	1.50	CD4081	.35
CD4006	1.75	CD4029	1.99	CD4082	.35
CD4007	.25	CD4030	.60	CD4510	1.95
CD4008	1.95	CD4033	2.49	CD4511	1.40
CD4009	.48	CD4034	4.95	CD4516	1.95
CD4010	.48	CD4035	1.79	CD4518	2.45
CD4011	.28	CD4040	2.10	CD4520	2.49
CD4012	.28	CD4042	1.45	CD4522	2.56
CD4013	.60	CD4043	1.45	CD4526	2.56
CD4014	1.95	CD4044	1.95	CD4527	2.70
CD4015	1.95	CD4049	.61	CD4528	1.50
CD4016	.59	CD4050	.61	CD4531	1.99
CD4017	1.25	CD4051	2.25	CD4543	2.25
CD4018	2.39	CD4066	1.20	CD4555	1.75
CD4019	1.95	CD4068	.35	CD4556	1.75
CD4020	1.35	CD4069	.35	CD4581	4.95
CD4021	1.85	CD4070	.99	CD4582	1.95
CD4023	.29	CD4071	.35	CD4585	2.45
CD4024	1.29	CD4072	.35	Also	
CD4025	.29	CD4073	.35	5101	$11.95

★ 7400 TTL

7400N	.18	7445N	.99	74122N	.44	74185N	$2.10
7401N	.18	7446N	.99	74123N	.60	74190N	$1.30
7402N	.18	7447N	.99	74125N	.79	74191N	$1.30
7403N	.18	7448N	.99	74126N	.79	74192N	$1.10
7404N	.19	7450N	.26	74132N	.70	74193N	$1.10
7405N	.20	7451N	.26	74141N	.88	74194N	.95
7406N	.25	7453N	.26	74145N	.70	74195N	.95
7407N	.25	7454N	.26	74147N	$1.59	74196N	.95
7408N	.21	7470N	.45	74148N	$1.29	74197N	.95
7409N	.21	7472N	.30	74150N	$1.15	74198N	$1.45
7410N	.21	7473N	.30	74151N	.98	74199N	$1.85
7411N	.21	7474N	.30	74153N	.98	74251N	$1.45
7412N	.21	7475N	.69	74154N	$1.15	74279N	$1.10
7413N	.45	7476N	.42	74155N	.95	8093N	.49
7414N	.69	7480N	.70	74156N	.95	8094N	.49
7416N	.35	7482N	.95	74157N	.95	8095N	.99
7417N	.35	7483N	.95	74160N	.99	8096N	.99
7420N	.19	7485N	.89	74161N	.99	8097N	.99
7421N	.35	7486N	.28	74162N	.99	8098N	.99
7423N	.35	7489N	$1.88	74163N	.99	75150N	$1.70
7425N	.35	7490N	.60	74164N	.99	75450N	$1.10
7426N	.35	7491N	.70	74165N	.99	75451N	.59
7427N	.35	7492N	.59	74166N	$1.10	75452N	.59
7428N	.28	7493N	.79	74170N	$4.20	75453N	.59
7430N	.21	7494N	.79	74172N	$16.25	75454N	.59
7432N	.31	7495N	.79	74173N	$1.40	75491N	.70
7433N	.30	7496N	.79	74174N	$1.25	75492N	.70
7437N	.35	7497N	$3.75	74175N	$1.25	75493N	$1.10
7438N	.35	74100N	$1.10	74176N	.79	75494N	$1.10
7440N	.21	74104N	.53	74177N	.79		
7441N	$1.05	74105N	.53	74180N	.79		
7442N	.69	74107N	.33	74181N	$2.75		
7443N	.69	74109N	.33	74182N	.79		
7444N	$1.50	74121N	.35	74184N	$2.10		

100 PIECES MIX TAKE 10% DISCOUNT
1000 PIECES MIX TAKE 15% DISCOUNT

PLESSEY SEMICONDUCTORS

RADIO COMMUNICATION

SL1610C	1.95	SL1624C	2.75
SL1611C	1.95	SL1630C	1.95
SL1612C	1.95	SL1640C	1.95
SL1613C	2.40	SL1641C	6.95
SL1620C	2.55	SL1645C	5.95
SL1621C	2.95	SL1650C	5.75
SL1622C	2.95	SL1651C	4.95
SL1623C	2.95	SL1680C	3.75

Application Manual for SL 600 Circuits 1.95
Plessey Semiconductor Catalogue 3.95

DIVIDERS

SP8600B	6.95	SP8620B	12.50	SP8640B	6.20
SP8601B	4.25	SP8621B	8.50	PS8641B	7.50
SP8602B	16.00	SP8622B	6.50	SP8642B	12.50
SP8603B	14.50	SP8630B	20.00	SP8643B	20.00
SP8604B	8.25	SP8631B	12.00	SP8646B	6.50
SP8607B	17.00	SP8632B	10.00	SP8647B	8.00
SP8613B	17.50	SP8634B	35.00	SP8650B	25.00
SP8614B	18.00	SP8635B	27.50	SP8651B	18.50
SP8615B	20.00	SP8638B	22.50	SP8652B	12.00
SP8616B	22.00	SP8637B	18.00		

ADVANCED ★ MICROCOMPUTER PRODUCTS
P. O. BOX 17329
IRVINE, CALIFORNIA 92713
PHONE (714) 968-3655

★ ORDER YOUR NEW CATALOG NOW !

Z-80 $49.95 4K STATIC RAM ★ 2708 $49.95

WIREWRAP SOCKETS - GOLD

14 Pin WW 1 level	.30
14 Pin WW 2 level	.33
14 Pin WW 3 level	.37
16 Pin WW 1 level	.31
16 Pin WW 2 level	.34
16 Pin WW 3 level	.38
18 Pin WW 3 level	.60
24 Pin WW 1 level	.70
24 Pin WW 2 level	.75
24 Pin WW 3 level	.85
40 Pin WW 3 level	1.25

EDGE CARD CONNECTORS

IMSA1 S100 WW .250" Row Spacing	$5.00
IMSA1 S100 ST .250" Row Spacing	5.00
Altair S100 WW Wire Wrap	5.50
Altair S100 ST Soldertail	5.25

D-SUBMIN CONNECTORS

RS-23225 Pin Male Part No. DB25P	$3.50
RS-23225 Pin Female Part No. DB25S	3.95
RS-232 Plastic Hood for above	.75

CRYSTALS — Microprocessors Timebases, TV Games Precision Crystals

Frequency	Case	Price	Frequency	Case	Price
1.000 MHz	HC6/HC33	4.95	13.000 MHz	HC18	4.95
2.000 MHz	HC6/HC33	5.85	14.31818 MHz	HC18	4.95
2.097162MHz	HC6	6.95	18.000 MHz	HC18	4.95
4.000 MHz	HC18	4.95	20.000 MHz	HC18	4.95
5.000 MHz	HC18	4.95	32.000 MHz	HC18	4.95
6.000 MHz	HC18	4.95	100Kc	HC13	12.95
10.000 MHz	HC18	4.95	3579.45 Kc	HC18	4.95
			2.4576 MHz		5.95

★ Exar

★ FUNCTION GENERATOR KIT ★

The Function Generator Kit features sine, triangle and square wave; THD 0.5% typ.; AM/FM capability.

XR-2206KA — Includes monolithic function generator IC, PC board, and assembly instruction manual — 18.95

★ TIMING CIRCUITS ★

XR555CP	Monolithic Timer	.90
XR556CP	Dual 555 Timer	1.75
XR2556CP	Dual Timing Circuit	2.95
XR2240CP	Programmable Timer	4.20
XR320P	Timer	1.40

★ COMMUNICATION CIRCUITS ★

XR1488	Quad RS232 Driver	1.45
XR1489	Quad RS232 Receiver	1.45
XR2208	Four Quadrant Multiplier	4.75

★ PHASE LOCKED LOOPS ★

XR2211CP	FSK Demod Tone Decoder	5.95
XR567CP	Tone Decoder	1.80
XR25567CP	Tone Decoder	1.80

★ STEREO DECODERS ★

XR1310P	FM Stereo Demodulator	3.00
XR1310EP	FM Stereo Demodulator	3.00
XR1800	Stereo Demodulator	3.00

★ FUNCTION GENERATORS ★

XR2206CP	Monolithic Function Generator	5.25
XR2207CP	Current Controlled Oscillator	3.75

★ OPERATIONAL AMPLIFIERS ★

XR742CP	Zero Crossing AC Trigger	3.00
XR4558CP	Dual Hi Gain Op Amp	1.20
XR4136CP	Quad Op Amp	1.75
XR4202CP	Programmable Quad Op Amp	4.25

2708 PRIME — ONLY $49.95

Build Your Own MONITOR
Now in Stock – NEW

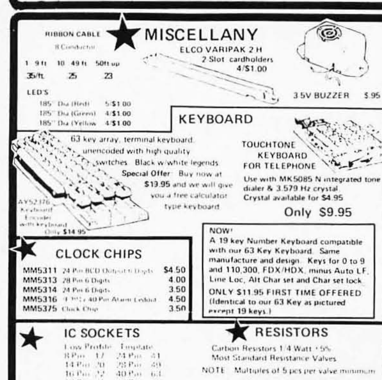
M3000 - 100 12-inch display module	$229.00
M2000 - 155 9-inch display module	$208.00

*Add $10.00 Shipping

MOTOROLA Exorciser, MEK6800DI and MEK6800D2 Compatible Modules

9601 16 Slot mother board - Inc. Conn.	$175.00
9602 16 Slot card cage 19" rack mount	$ 72.00
9610 Utility Prototyping Board	$ 36.00
9615 4K Eprom Module (1702A)	$350.00
9620 Parallel I/O Module	$375.00
9626 8K Static Ram Module	$350.00
9650 8 Channel Duplex Ansyn I/O	$390.00

All assembled and tested. Not Kits



★ MISCELLANY

RIBBON CABLE
8 Conductor
1-9 ft .35/ft — 10-49 ft .25 — 50 ft up .23

ELCO VARIPAK 2-H
2 Slot cardholders
4/$1.00

3.5V BUZZER $.95

LED'S
185" Dia (Red) 5/$1.00
185" Dia (Green) 4/$1.00
185" Dia (Yellow) 4/$1.00

KEYBOARD

63 key array, terminal keyboard, unencoded with high quality switches. Black w/white legends. Special Offer: Buy now at $19.95 and we will give you a free calculator type keyboard.

AY52376 Keyboard Encoder with keyboard Only $14.95

TOUCHTONE KEYBOARD FOR TELEPHONE
Use with MK5085 N integrated tone dialer & 3.579 Hz crystal.
Crystal available for $4.95
Only $9.95

NOW!
A 19 key Number Keyboard compatible with our 63 Key Keyboard. Same manufacture and design. Keys for 0 to 9 and 110,300, FDX/HDX, minus Auto LF, Line Loc, Alt Char set and Char set lock.
ONLY $11.95 FIRST TIME OFFERED
(Identical to our 63 Key as pictured except 19 keys.)

★ CLOCK CHIPS

MM5311	24 Pin BCD Outputs 6 Digits	$4.50
MM5313	28 Pin 6 Digits	4.00
MM5314	24 Pin 6 Digits	3.50
MM5316	40 Pin Alarm Ledout	4.50
MM5375	Clock Chip	3.50

★ IC SOCKETS

Low Profile Template

8 Pin	.17	24 Pin	.41
14 Pin	.20	28 Pin	.49
16 Pin	.22	40 Pin	.63
18 Pin	.31	22 Pin	.35

★ RESISTORS

Carbon Resistors 1/4 Watt ±5%
Most Standard Resistance Valves
NOTE: Multiples of 5 pcs per valve minimum

Qty (each valve)	Price
5-20	.05 each
25-50	.045 each
55-100	.04 each
105-500	.03 each
505 up	.025 each

★ VARIABLE RESISTORS

Centralab RN4 NAYSD503A (HMP 50K) $1.95
RN4 NAYSD503A (53C3 50K S) $1.95

★ CORCOM EMI FILTER

- CORCOM 2B4 2A • Only $4.95 — 115/230V 50-400Hz Test 2100VDC
- CORNELL DUBILIER NF21015 — 2x4A 250 VAC Only $4.95
- DIP SWITCH — 8 SPST Slide Switches in Dip $2.25

ROTARY SWITCH

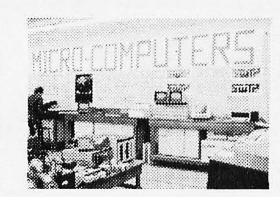
Centralab PA 2001 $1.95



C&K V 25 Bat handle toggle switch $1.50







COMPUTER WAREHOUSE STORE

DEPT: B • 584 COMMONWEALTH AVENUE • BOSTON, MA • 02215 • 617-261-2701 • VISIT US: 9-9 WEEKDAYS; 9-6 SATURDAYS

ONE DAY SHIPMENT

PERIPHERALS FOR MICROSYSTEMS

ALL an ASR 33 is and MORE!



RS232 INTERFACE, QUIET OPERATION, 10 CPS PRINTER, BUILT-IN PAPER TAPE READER/PUNCH, ELECTRIC TYPEWRITER KEYBOARD WITH ADDITIONAL 10 KEY NUMERIC PAD, YOUR CHOICE OF FRICTION OR SPROCKET FEED, LIGHTED PLATEN AREA FOR EASY READING, STANDARD PAPER AND TAPE, SUPPORTED BY OLIVETTI.

$950 + SHIPPING 165 lb

TALLY T132



7 x 8 DOT MATRIX IMPACT PRINTER HAS A SINGLE LINE DYNAMIC MEMORY AND A UNIVERSAL INTERFACE TO ACCEPT PARALLEL DATA, FORMS TO 14-7/8 IN. WIDE, SIMPLE PRINTING MECHANISM USES 132 SOLENOID HAMMERS AND TWO STEPPER MOTORS FOR 100 LPM, 132 COLUMNS, 64 CHARACTERS

$450 + SHIPPING 150 lb

DIGITRONICS D507.....$95 + 400 lb SHIPPING

PAPER TAPE READER, HEAVY DUTY PS, 3 MUFFIN FANS, POWER CONTROL PANEL, CIRCUIT BOARDS, RELAYS IN CABINET

KDI ADTROL AR-21...$95 + $25 SHIPPING

ELECTRO OPTICAL PAPER TAPE READER WITH 110V PS, STEPPER MOTOR, 250 CHAR/SEC, FAN IN TABLE TOP HOUSING

IBM 731 I/O WRITER $750 + $25 SHIPPING

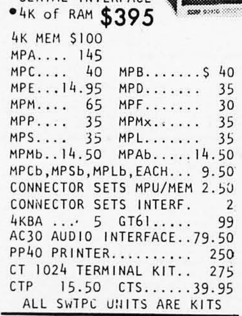
8½"PLATEN,PINFD,EBCDIC,U/L CASE,DUAL CLR RIBBON,115B

KITS ★ HIGHLIGHTS FROM OUR WIDE SELECTION ★

IMSAI 8080 MICROKITS

8080A KIT 5 SLOT,......$599
8080A KIT 22 SLOT...... 651
4K MEMORY KIT.......... 139
PIC-8 PRIORITY INTERRPT 125
SERIAL I/O KIT......... 125
PROM 4-512 KIT......... 165
UCRI-1 KIT............. 59
CABLE A KIT............ 18
STANDARD INPUT/OUT INTERFACES, MANUALS, SOCKET SETS

SWTPC 6800



- 512 BYTES of ROM
- RS232 or 20 mA
- SERIAL INTERFACE
- 4K of RAM **$395**

4K MEM $100
MPA.... 145
MPC.... 40 MPB.......$ 40
MPE...14.95 MPD....... 35
MPM.... 65 MPF....... 30
MPP.... 35 MPMx...... 35
MPS.... 35 MPL....... 35
MPMb..14.50 MPAb.....14.50
MPCb,MPSb,MPLb,EACH... 9.50
CONNECTOR SETS MPU/MEM 2.50
CONNECTOR SETS INTERF. 2
4KBA ... 5 GT61..... 99
AC30 AUDIO INTERFACE..79.50
PP40 PRINTER.......... 250
CT 1024 TERMINAL KIT.. 275
CTP 15.50 CTS......39.95
ALL SWTPC UNITS ARE KITS

VIKING 100 PIN CONNECTOR HEAVY DUTY $ 3

LEAR SIEGLER ADM-3A W/ Cursor Control

- 12" CRT
- 24 LN X 80 CHAR
- RS232
- 20 mA LOOP **$875**

ICOM MICROFLOPPIES

PLUG COMPATIBLE FOR:
S100 BUS...FD2411..... **$1095**
SINGLE DRIVE FD2402......$ 649
ICOM FLOPPIES:
FF36-1 FRUGAL..........$1195
FF36-2 DUAL FRUGAL..... 1895
360-58 BLT;INTFC 8080.. 300
S171 POWER SUPPLY ... 250
FD360-2-5DUAL SYSTEM... 3000

KIM-1 6502.................$ 245
KIM-2 4K................. 179
KIM-3 8K................. 289
MANUALS PACKAGE.......... 15

TARBELL AUDIO CASSETTE KIT 120

INTERSIL INTERCEPT JR. $281

12K RAM.................. 145
ROM/PROM BOARD.........74.65
YOU ADD MEMORY CHIPS
SERIAL I/O............81.50
AUDIO VISUAL BOARD..... 125

SCAMP KIT...............$99

FROM NATIONAL SEMICONDUCTOR
KEYBOARD KIT.......... 95

SMOKE SIGNAL BROADCASTING, 16K RAM.............. 595

SPECIAL DISCOUNTS! ON KITS & ASSEMBLED UNITS

SAVE UP TO 20% OFF KIT PRICE WHEN A PERIPHERAL IS PURCHASED AT THE SAME TIME.

PERIPHERAL PRICE OVER $900 ➡ 20% OFF KIT
PERIPHERAL PRICE OVER $250 ➡ 10% OFF KIT
PERIPHERAL PRICE OVER $95 ➡ 5% OFF KIT

GREEN PHOSPHOR VIDEO MONITOR $150 +25 SHIPPING

STANDARD 1V P TO P COMPOSITE VIDEO INPUT, 16 MHz BAND WIDTH, RASTER SCAN 12x12x13" WITH POWER SUPPLY, VIDEO AMPLIFIER, DRIVING CIRCUITRY VENTILATION MUFFIN FANS, 7x9" HORIZONTAL VIEWING, CAPABLE OF 24 LINES x 80 CHAR., ANTIGLARE ¼" ETCHED GRADIENT DENSITY FACE PLATE, P39 GREEN PHOSPHOR, ON/OFF BRIGHTNESS CONTROLS, 115Vac, 60 W ... TRULY A COMMERCIAL UNIT BUILT TO WORK IN A DEMANDING ENVIRONMENT.

DATAPOINT 3300-200 THERMAL PRINTER

PARALLEL PRINTER WITH ADDITIONAL CIRCUIT BOARDS TO PROVIDE SERIAL INTERFACE, PRINTS UP TO 30 CPS, 100Vac PS USES WIDELY AVAILABLE NCR PAPER, 96 CH, ASCII, 80 COL, CRT COMPATIBLE 5x7 DOT MATRIX, SOLID STATE WITH LESS THAN 25 MOVING PARTS.

$475 + $25 SHIPPING

DATAPOINT CONSOLE PRINTER

BOTH UNIVAC AND SINGER BUILT THESE PRINTER MECHANISMS WHICH OPERATE AT 30 CPS FROM A ROTATING WHEEL. 65 CHAR. USES STANDARD FORM FEED PAPER. PINWHEEL IS INTERCHANGEABLE. **$395** + SHIPPING 285 lb.

UNIVAC 0769-06 PRINTER MECHANISM ONLY ... $195 INCLUDES MOTOR/PRINT WHEEL +SHIPPING 75lb.

KLEINSCHMIDT 311.....$250 + 75 lb SHIPPING

THIS 30 CH/SEC DRUM PRINTER SITS IN A SOUND-PROOF ENCLOSURE. 64 CHAR, PARALLEL INPUT, 80 CHAR/LN

TECHTRAN 4100........$595 + 35 lb SHIPPING

TAPE CASSETTE DRIVE. CAN RUN DIRECTLY FROM TERMINAL INDEPENDENT OF CPU. FULL EDIT CAPABILITY.

TO ORDER EQUIPMENT

1. ENCLOSE CHECK FOR FULL PRICE PLUS SHIPPING CHARGES (KITS - ADD $5 IF UNDER $100; $10 IF OVER) BANKAMERICARD & MASTERCHARGE ACCEPTED SEND CARD NUMBER, EXPIRATION DATE, INTERBANK #
2. CLEARLY IDENTIFY SHIPPING ADDRESS
3. DESCRIBE ITEM BY MODEL NUMBER

ALL MERCHANDISE WARRANTEED

WRITE FOR OUR CATALOGUE

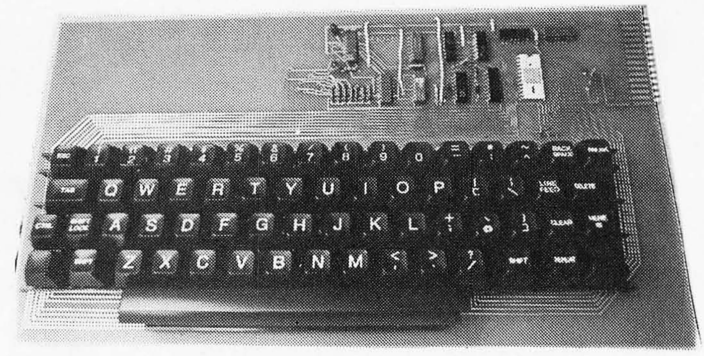
INCLUDES DESCRIPTIONS OF OUR COMPLETE LINE OF KITS & UNITS, REVIEWS OF OVER 150 BOOKS, LIST OF NEW & SURPLUS PARTS AND ARTICLES - "ALL ABOUT PERSONAL COMPUTERS" **$1**

THE SC/MP AT $ 95.00

SC/MP, the Microprocessor kit from **National Semiconductor** includes everything you need to build a completely functional microprocessor system — featuring the National SC/MP microprocessor — the low cost microprocessor for every application: Text Systems and Instrument Control; Machine Tool Control; Small Business Machines; Word Processing Systems; Educational Systems; Multiprocessor Systems; Proces Controllers; Terminal Control; Laboratory Instrumentation; Sophisticated Games; Automotive Controller and Appliance Controllers.

The kit, neatly packaged with all the components and literature you need, in a looseleaf binder, includes: The SC/MP Microprocessor — a single-chip Central Processing Unit in a 40-pin, dual in-line package. Features static operations, forty-six instruction types; single-byte and double-byte, software controlled interrupt structure, built in serial input/output ports; bidirectional 8-bit TRI-STATE[R] bus, parrallel data/port and latched 12-bit TRI-STATE[R] address port. ROM — 512 bytes (8-bits/byte) of pre-programmed Read-Only-Memory containing KITBUG—a monitor and debugging program to assist in the development of your application programs, KITBUG provedes teletypewrite input/output routines and allows examination, modification, and controlled execution of your programs. RAM-256 bytes of static read/write memory for storage of your application programs. Transfers of data to and from RAM are controlled by SC/MP and KITBUG. Teletypewriter Interfact including buffer and drive capability for a 20 MA current loop interface. Voltage Regulator. Data Buffer—providing interface between memory and bidirectional data lines. All the literature you need, including schematics and programming manuals. Timing Crystal—providing 1.000 MHz timing signal. Plus all the passive components and circuit board with 72 pin edge connector required to build and interconnect your microprocessor system with external hardware.

ASC II KEYBOARD



(Reg. $58.85) **$53.00**

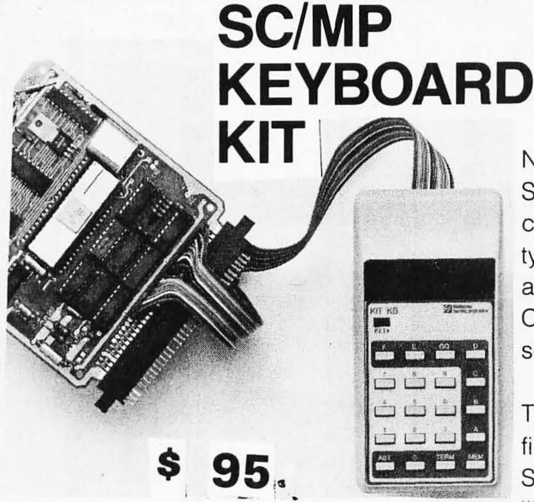
This 63 key **ASC II Encoded Keyboard** kit was designed and manufactured by **Electronics Warehouse Inc.** Features: single 5 volt D.C. supply, utilizing only TTL logic elements (no MOS devices to blow), TTL drive capability (each of the eight bits of ASC II output will drive the equivalent of ten standard TTL inputs without external buffer drivers), de-bouncing, upper and lower case fully ASC II, 8 bit parallel output. In addition to the alpha-numeric and symbol keys available on a regular keyboard, the following keys are utilized: escape, back-space, tab, line-feed, delete, control, shift-lock, shift (2 keys), here-is, control-release.

Kit includes: 63 key keyboard, P.C. board, all required components and assembly manual with ASC II code list.

Optional: 1. Parity bit - add $1.00 **Aluminum enclosure $39.–**
2. Serial output - add $2.00

Note: If you already have this teletype keyboard you can have the kit without it for **$36.00 (reg. $39.85).** Dealer inquires invited.

SC/MP KEYBOARD KIT



This is a great kit for engineers and companies who don't have access to a Teletype. It is a low-cost teaching, learning, and developing tool for hobbyists, professors, students, and electronics entrepreneurs of all levels.

National's new Keyboard Kit now gives SC/MP Kit users a low-cost input/output capability. This new kit replaces the Teletype* normally required by the SC/MP Kit and allows users to evaluate the SC/MP CPU and to develop a variety of application software.

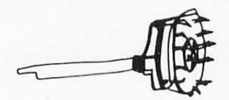
The heart of SC/MP Keyboard Kit is a ROM firmware package (512 bytes) called SCMPKB. The SCMPKB ROM replaces the "Kit Bug" ROM originally supplied with the SC/MP Kit and allows the effective use of the hexadecimal keyboard, to execute programs, to examine or modify the contents of memory and the SC/MP registers, and to monitor program performance.

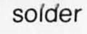
There is a hole pattern for additional integrated circuits on the SC/MP Kit PC card. By following the simple instructions in the SC/MP Keyboard Kit users manual, one can add buffers, decoders, drivers, multiplexers, etc. Simply replace the Kit Bug ROM (supplied in the SC/MP Kit) with the new SCMPKB ROM, connect the pre-assembled Keyboard cable connector to the kit card, and you are ready to go!

National's Keyboard Kit comes complete with manual, all required integrated circuits, resistors, keyboard display cable connector assembly, wire wrap connectors, precut wires—even a hand-held wire wrap tool.

SWITCHES

10 Position rotary switch
by oak manuf. $1.00 per 4





8 position, 1 off switch in TO5 can
$.69 2 for $1.00

ROCKER SWITCH

SPST normally open contact rating 6A 250VAC
solder **$ .50 ea $2.00 per 6**

MINIATURE TOGGLE SWITCH
SPST P.C. MOUNT **$.99**



MINIMUM ORDER $5.00

Shipping & Handling:

KEYBOARD or SC/MP - $3.00 + $.50 Insurance

all others - $1.00

California residents add 6% sales tax.

ELECTRONICS WAREHOUSE Inc.

1603 AVIATION BLVD.
REDONDO BEACH, CA. 90278
TEL. (213) 376-8005

WRITE FOR FREE CATALOG

You are invited to visit our store at the above address

7400N TTL

SN7400N	.16	SN7459A	.25		
SN7401N	.16	SN7460N	.22	SN74154N	1.00
SN7402N	.21	SN7470N	.45	SN74155N	.99
SN7403N	.16	SN7472N	.39	SN74156N	.99
SN7404N	.18	SN7473N	.37	SN74157N	.99
SN7405N	.24	SN7474N	.32	SN74160N	1.25
SN7406N	.20	SN7475N	.50	SN74161N	.99
SN7407N	.29	SN7476N	.32	SN74163N	.99
SN7408N	.25	SN7479N	5.00	SN74164N	1.10
SN7409N	.25	SN7480N	.50	SN74165N	1.10
SN7410N	.18	SN7482N	.98	SN74166N	1.25
SN7411N	.30	SN7483N	.70	SN74167N	5.50
SN7412N	.33	SN7485N	.89	SN74170N	2.10
SN7413N	.45	SN7486N	.39	SN74172N	8.95
SN7414N	.70	SN7488N	3.50	SN74173N	1.50
SN7416N	.35	SN7489N	2.49	SN74174N	1.25
SN7417N	.35	SN7490N	.45	SN74175N	.99
SN7420N	.21	SN7491N	.75	SN74176N	.90
SN7421N	.33	SN7492N	.49	SN74177N	.90
SN7422N	.49	SN7493N	.49	SN74180N	.99
SN7423N	.37	SN7494N	.79	SN74181N	2.49
SN7425N	.29	SN7495N	.79	SN74182N	.95
SN7426N	.29	SN7496N	.89	SN74184N	1.95
SN7427N	.37	SN7497N	4.00	SN74185N	2.20
SN7429N	.42	SN74100N	1.00	SN74186N	15.00
SN7430N	.26	SN74107N	.39	SN74187N	6.00
SN7432N	.31	SN74121N	.39	SN74188N	3.95
SN7437N	.27	SN74122N	.39	SN74190N	1.19
SN7438N	.27	SN74123N	.50	SN74191N	1.25
SN7439N	.25	SN74125N	.60	SN74192N	.89
SN7440N	15	SN74126N	.60	SN74193N	.89
SN7441N	.89	SN74132N	1.09	SN74194N	1.25
SN7442N	59	SN74136N	.95	SN74195N	.75
SN7443N	.75	SN74141N	1.15	SN74196N	1.25
SN7444N	.75	SN74142N	4.00	SN74197N	.75
SN7445N	.75	SN74143N	4.50	SN74198N	1.75
SN7446N	81	SN74144N	4.50	SN74199N	1.75
SN7447N	.69	SN74145N	1.15	SN74200N	5.59
SN7448N	79	SN74147N	2.35	SN74279N	.90
SN7450N	26	SN74148N	2.00	SN74251N	1.79
SN7451N	27	SN74150N	1.00	SN74284N	6.00
SN7453N	.27	SN74151N	.79	SN74285N	6.00
SN7454N	20	SN74153N	.89	SN74367N	.75

MANY OTHERS AVAILABLE ON REQUEST
20% Discount for 100 Combined 7400's

CMOS

CD4000	.25			74C04N	.75
CD4001	.25			74C10N	.65
CD4002	.25	CD4035	1.85	74C20N	.65
CD4006	2.50	CD4040	2.45	74C30N	.65
CD4007	.25	CD4042	1.90	74C42N	2.15
CD4009	.59	CD4044	1.50	74C73N	1.50
CD4010	.59	CD4046	2.51	74C74	1.15
CD4011	.25	CD4047	2.75	74C90N	3.00
CD4012	.25	CD4049	.79	74C95N	2.00
CD4013	.47	CD4050	.79	74C107N	1.25
CD4016	.56	CD4051	2.95	74C151	2.90
CD4017	1.35	CD4053	2.95	74C154	4.00
CD4019	.55	CD4060	3.25	74C157	2.15
CD4020	1.49	CD4066	1.75	74C160	3.25
CD4022	1.25	CD4069	.45	74C161	3.25
CD4023	.25	CD4071	.45	74C163	3.00
CD4024	1.50	CD4081	.45	74C164	3.25
CD4025	.25	CD4511	2.50	74C173	2.60
CD4027	.69	CD4518	2.50	74C193	2.75
CD4028	1.65	MC14566	3.00	74C195	2.75
CD4029	2.90	74C00N	.39	MC4044	4.50
CD4030	.65	74C02N	.55	MC14016	.56

LINEAR

LM300H	.80			LM1310N	2.95
LM301H	.35			LM1351N	1.65
LM301CN	.35			LM1414N	1.75
LM302H	.75			LM1458C	.65
LM304H	1.00	LM370N	1.1.5	LM1496N	.95
LM305H	.95	LM373N	3.25	LM1556V	1.85
LM307CN	.35	LM377N	4.00	LM2111N	1.95
LM308H	1.00	LM380N	1.39	LM2901N	2.95
LM308CN	1.00	LM380CN	1.05	LM3065N	.69
LM309H	1.10	LM381N	1.79	LM3900N	.55
LM309K	.99	LM382N	1.79	LM3905N	.60
LM310CN	1.15	NE501K	8.00	LM3909	1.25
LM311H	.90	NE510A	6.00	LM5556N	1.85
LM311N	.90	NE531H	3.00	MC5558V	1.00
LM318CN	1.50	NE536T	6.00	LM7525N	.90
LM319N	1.30	NE540L	6.00	LM7535N	1.25
LM320K-5	1.35	NE550N	.79	8038B	4.95
LM320K-5.2	1.35	NE555V	.39	LM75450	.49
LM320K-12	1.35	NE560B	5.00	75451CN	.39
LM320K-15	1.35	NE561B	5.00	75452CN	.39
LM320T-5	1.75	NE562B	5.00	75453CN	.39
LM320T-5.2	1.75	NE565H	1.25	75454CN	.39
LM320T-8	1.75	NE565N	1.75	75491CN	.79
LM320T-12	1.75	NE566CN	1.25	75492CN	.89
LM320T-15	1.75	NE567H	1.95	75494CN	.89
LM320T-18	1.75	NE567V	1.50	**RCA LINEAR**	
LM320T-24	1.75	LM703CN	.45	CA3013	2.15
LM323K-5	9.95	LM709H	.29	CA3032	2.56
LM324N	1.80	LM709N	.29	CA3035	2.48
LM339N	1.70	LM710N	.79	CA3039	1.35
LM340K-5	1.95	LM711N	.39	CA3046	1.30
LM340K-6	1.95	LM723H	.55	CA3059	3.25
LM340K-8	1.95	LM723H	.55	CA3060	3.25
LM340K-12	1.95	LM733N	1.00	CA3080	.85
LM340K-15	1.95	LM739N	1.00	CA3081	2.00
LM340K-18	1.95	LM741CH	.35	CA3082	2.00
LM340K-24	1.95	LM741CN	.35	CA3083	1.60
LM340T-5	1.75	LM741-14N	.39	CA3086	.85
LM340T-6	1.75	LM747H	.79	CA3089	3.75
LM340T-8	1.75	LM747N	.79	CA3091	10.20
LM340T-12	1.75	LM748H	.39	CA3123	2.15
LM340T-15	1.75	LM748N	.39	CA3130	1.39
LM340T-18	1.75	LM1303N	.90	CA3140	1.25
LM340T-24	1.75	LM1304N	1.19	CA3600	1.75
LM350N	1.00	LM1305N	1.40	RC4194	5.95
LM351CN	.65	LM1307N	.85	RC4195	3.25

74LS00 TTL

74LS00	29			74LS139	1.95
74LS02	29			74LS151	1.55
74LS03	29	74LS74	49	74LS153	1.89
74LS04	35	74LS75	69	74LS157	1.55
74LS05	35	74LS76	49	74LS162	2.25
74LS08	29	74LS83	1.75	74LS163	2.25
74LS10	29	74LS85	2.49	74LS164	1.95
74LS13	69	74LS86	49	74LS175	1.95
74LS14	1.75	74LS90	1.25	74LS181	3.69
74LS20	29	74LS92	1.25	74LS190	2.85
74LS26	39	74LS93	1.25	74LS191	2.85
74LS27	39	74LS95	1.95	74LS192	2.85
74LS28	39	74LS96	1.89	74LS193	2.85
74LS30	29	74LS107	.59	74LS194	1.89
74LS32	39	74LS109	.59	74LS195	1.89
74LS40	39	74LS112	.59	74LS257	1.75
74LS51	29	74LS132	1.25	74LS260	55
74LS55	29	74LS136	.59	74LS279	79
74LS73	49	74LS138	1.89	74LS670	3.95

CLOCK CHIPS

MM5309	6 Digit, BCD Outputs, Reset PIN.	$9.95
MM5311	6 Digit, BCD Outputs, 12 or 24 Hour	4.95
MM5312	4 Digit, BCD Outputs, 1 PPS Output	4.95
MM5314	6 Digit, 12 or 24 Hour, 50 or 60 Hz	4.95
MM5316	4 Digit, Alarm, 1 PPS Output	6.95
MM5318	Video Clock Chip, For Use With (MM5841 - $9.95)	9.95
CT7001	6 Digit, Calendar, Alarm, 12 or 24 Hour	5.95

DATA HANDBOOKS

7400	Pin-out & Description of 5400/7400 ICS	$2.95
CMOS	Pin-out & Description of 4000 Series ICS	$2.95
Linear	Pin-out & Functional Description	$2.95

ALL THREE HANDBOOKS $6.95

FAIRCHILD TECHNOLOGY KITS FAIRCHILD

DIGITS

FTK0001	0.5" High Common Cathode Digit	$1.00
FTK0002	0.5" High Common Anode Digit	1.00
FTK0003	.357" High Common Cathode Digit	.75
FTK0004	0.8" High Common Cathode Digit	2.00
FTK0005	0.8" High Common Anode Digit	2.00

0.8" HIGH DISPLAY ARRAYS

FTK0010	12 Hour, 3½ Digit Clock Display	7.00
FTK0011	24 Hour, 4 Digit Clock Display	8.00

LED LAMPS

FTK0020	10 Red LED Lamps	1.00
FTK0021	5 Mixed Colored LED Lamps	1.00
FTK0022	10 LED Mounting Clips	1.00
FTK0023	5 Three Piece LED Mounting Adapters	1.00

PHOTO TRANSISTORS

FTK0030	5 Flat Lens Photo Transistors	1.00
FTK0031	5 Round Lens Photo Transistors	1.00
FTK0032	3 Flat Lens Photo Darlingtons	1.00
FTK0033	3 Round Lens Photo Darlingtons	1.00

PHOTO ARRAYS

FTK0040	9-Element Tape Reader Array	16.00
FTK0041	12-Element Card Reader Array	24.00
FTK0042	Reflective Opto Coupler	4.00

COUPLERS

FTK0050	3 General Purpose Opto Couplers	1.00
FTK0051	Darlington Opto Coupler	1.00

MOS CLOCK CIRCUITS

FTK0400	Digital Clock/Calendar Circuit (FCM7001)	7.00
FTK0401	Digital Clock/Calendar with BCD Outputs (FCM7002)	7.00
FTK0402	Direct Drive Digital Clock Circuit with AC Output (FCM3817A)	5.00
FTK0403	Direct Drive Digital Clock Circuit with DC Output (FCM3817D)	5.00
FTK0405	Direct Drive Digital Clock/Calendar Circuit (FCM7015)	6.00

KITS

FTK0106	Automobile Clock Kit	40.00

DISCRETE LEDS

125" dia.			.190" dia.		
XC209	Red	10/$1	XC111	Red	10/$1
XC209	Green	4/$1	XC111	Green	4/$1
XC209	Orange	4/$1	XC111	Yellow	4/$1
XC209	Yellow	4/$1	XC111	Orange	4/$1

.200" dia			.185" dia.			.200" dia.		
XC22	Red	10/$1	XC526	Red	10/$1	XC556	Red	10/$1
XC22	Green	4/$1	XC526	Green	4/$1	XC556	Green	4/$1
XC22	Yellow	4/$1	XC526	Yellow	4/$1	XC556	Yellow	4/$1
XC22	Orange	4/$1	XC526	Orange	4/$1	XC556	Orange	4/$1
SSL-22	RT	4/$1	XC526	Clear	4/$1	XC556	Clear	7/$1

.085" dia.
MV50 — 6/$1

INFRA-RED LED
¼" x ¼" x 1/16"
Flat **5/$1.00**

SPECIAL * — XC556 Red 100/$8.00 1000/$60.00 — SPECIAL *

DISPLAY LEDS

DL707 DL33B

TYPE	POLARITY	HT		TYPE	POLARITY	HT	
MAN 1	Common Anode	270	2.95	MAN 3640	Common Cathode-orange	300	1.75
MAN 2	5 x 7 Dot Matrix	300	4.95	MAN 4710	Common Anode-Red	400	1.95
MAN 3	Common Cathode	125	3/1.00	DL701	Common Anode-red ±	300	99
MAN 4	Common Cathode	187	1.95	DL704	Common Cathode	300	99
MAN 7	Common Anode	300	1.25	DL707	Common Anode	300	99
MAN 7G	Common Anode-green	300	1.95	DL 728	Common Cathode	500	1.95
MAN 7Y	Common Anode-yellow	300	1.95	DL741	Common Anode	600	1.50
MAN 52	Common Anode-green	300	99	DL 747	Common Anode	600	2.25
MAN 64	Common Anode-red	400	99	DL 750	Common Cathode	600	2.49
MAN 74	Common Cathode	300	1.50	DL 33B	Common Cathode	110	.50
MAN 82	Common Anode-yellow	300	99	FND70	Common Cathode	250	.75
MAN 84	Common Cathode-yellow	300	99	FND503	Common Cathode	500	1.00
MAN 3620	Common Anode-orange	300	1.75	FND507	Common Anode	500	1.00

FACTORY REJECTS ATARI GAME BOARDS

BOARD A — 8½" x 16"
Over 60 each reusable IC's
Misc. Transistors, Resistors,
Diodes, Caps, Crystals, Switch, etc.
$6.95 ea. ONLY 500 EA AVAILABLE

DELUXE BOARD B — 11½" x 18"
Over 100 each reusable IC's
Misc. Tranistors, Resistors,
Diodes, Caps, Crystals, Switches, LEDS, etc.
$9.95 ea. ONLY 500 EA AVAILABLE

NEW AY-3-8500-1 TV GAME CHIP

- Score display
- Selectable angles, speed and bat size
- Auto or manual ball service
- Realism sounds
- 6 Games — tennis, hockey/soccer, squash, pelota/practice and two rifle games

28 Lead Dual In Line
The circuit is intended to be battery powered. Minimum number of external components required for complete system.

$24.95

IC SOLDERTAIL — LOW PROFILE (TIN) SOCKETS

	1-24	25-49	50-100		1-24	25-49	50-100
8 pin	$.17	.16	.15	24 pin	$.38	.37	.36
14 pin	.20	.19	.18	28 pin	.45	.44	.43
16 pin	.22	.21	.20	36 pin	.60	.59	.58
18 pin	.29	.28	.27	40 pin	.63	.62	.61
22 pin	.37	.36	.35				

SOLDERTAIL STANDARD (TIN)

14 pin	$.27	.25	.24	28 pin	$ .99	.90	.81
16 pin	.30	.27	.25	36 pin	1.39	1.26	1.15
18 pin	.35	.32	.30	40 pin	1.59	1.45	1.30
24 pin	.49	.45	.42				

SOLDERTAIL STANDARD (GOLD)

8 pin	$.30	.27	.24	24 pin	$ .70	.63	.57
14 pin	.35	.32	.29	28 pin	1.10	1.00	.90
16 pin	.38	.35	.32	36 pin	1.75	1.40	1.26
18 pin	.52	.47	.43	40 pin	1.75	1.59	1.45

WIRE WRAP SOCKETS (GOLD) LEVEL #3

10 pin	$.45	.41	.37	24 pin	$1.05	.95	.85
14 pin	.39	.38	.37	28 pin	1.40	1.25	1.10
16 pin	.43	.42	.41	36 pin	1.59	1.45	1.30
18 pin	.75	.68	.62	40 pin	1.75	1.55	1.40

Plastic Push Button Switch

- 18 AWG Solid Wire - 5" Long
- .50 (wide) X .60 (high) ⅜-27 Thread
- 8 AMP @ 14 Volt - 1 AMP @ 110 Volt

		1-9	10-Up
J-188-1	Push On-Push Off	.59	.49
J-188-2	Normally Open	.59	.49
J-188-3	Normally Closed	.59	.49

MINATURE TOGGLE SWITCH

JMT-221	DPDT	on/off/on	$1.95
JMT-223	DPDT	on/none/on	$1.75
JMT-121	SPDT	on/off/on	$1.50
JMT-123	SPDT	on/none/on	$1.25

CLIPLITE 8/$1.49

LED MOUNTING SYSTEM
use with XC556 LEDS
Specify Colors — Red - Green - Amber - Yellow

DIP SWITCH

These switches feature 7 SPST slide switches in a molded dip. They are ideally suited for microprocessor applications. **$1.95**

50 PCS. RESISTOR ASSORTMENTS $1.75 PER ASST.

		10 OHM	12 OHM	15 OHM	18 OHM	22 OHM	
ASST. 1	5 ea.	27 OHM	33 OHM	39 OHM	47 OHM	56 OHM	1/4 WATT 5% = 50 PCS.
		68 OHM	82 OHM	100 OHM	120 OHM	150 OHM	
ASST. 2	5 ea.	180 OHM	220 OHM	270 OHM	330 OHM	390 OHM	1/4 WATT 5% = 50 PCS.
		470 OHM	560 OHM	680 OHM	820 OHM	1K	
ASST. 3	5 ea.	1.2K	1.5K	1.8K	2.2K	2.7K	1/4 WATT 5% = 50 PCS.
		3.3K	3.9K	4.7K	5.6K	6.8K	
ASST. 4	5 ea.	8.2K	10K	12K	15K	18K	1/4 WATT 5% = 50 PCS.
		22K	27K	33K	39K	47K	
ASST. 5	5 ea.	56K	68K	82K	100K	120K	1/4 WATT 5% = 50 PCS.
		150K	180K	220K	270K	330K	
ASST. 6	5 ea.	390K	470K	560K	680K	820K	1/4 WATT 5% = 50 PCS.
		1M	1.2M	1.5M	1.8M	2.2M	
ASST. 7	5 ea.	2.7M	3.3M	3.9M	4.7M	5.6M	1/4 WATT 5% = 50 PCS.

ASST. 8R Includes Resistor Assortments 1-7 (350 PCS.) **$10.95 ea.**

$5.00 Minimum Order — U.S. Funds Only
California Residents — Add 6% Sales Tax

Spec Sheets - 25¢ — Send 35¢ Stamp for 1977A Catalog
Dealer Discount Available — Request Pricing

NOW OPEN SATURDAYS

James ELECTRONICS

1021-A HOWARD AVE., SAN CARLOS, CA. 94070
PHONE ORDERS WELCOME — (415) 592-8097
All Advertised Prices Good Thru April

WIRE WRAP CENTER

HOBBY-WRAP TOOL-BW-630

- Battery Operated (Size C)
- Weighs ONLY 11 Ounces
- Wraps 30 AWG Wire onto Standard DIP Sockets (0.25 inch)
- Complete with built-in bit and sleeve

$34.95
(batteries not included)

WIRE-WRAP KIT — WK-2-W

WRAP • STRIP • UNWRAP

- Tool for 30 AWG Wire
- Roll of 50 Ft. White or Blue 30 AWG Wire
- 50 pcs. each 1", 2", 3" & 4" lengths — pre-stripped white wire

$11.95

WIRE WRAP TOOL WSU-30

WRAP • STRIP • UNWRAP - $5.95

WIRE WRAP WIRE — 30 AWG

25 ft. min. $1.25 50 ft. $1.95 100 ft. $2.95 1000 ft. $15.00
SPECIFY COLOR — White - Yellow - Red - Green - Blue - Black

CUTTER CRIMPER TOOL (CS-8)

- Plier Nose (serrated-jaw)
- Scissors Action Cutting
- 6 Bolt Cutters (4-40, 5-40, 6-32, 8-32, 10-32, 10-24)
- Crimp Stations (7mm Auto — 22-20 to 12-X0 elect.)
- "Up-Front" Wire Cutting
- Scissors Action Stripping (No. 22-20 to No. 10)
- Crimp Stations — insulated (2w-20 to 12-10 elect.)

Actual Size - 8¼" length **$8.50**

Permacel Electrical Tape

- ¾" (wide) X 66 ft. (long) • All weather • Not import

$1.25 per roll — $9.95 per 10 roll package

ZENERS — DIODES — RECTIFIERS

TYPE	VOLTS	W	PRICE	TYPE	VOLTS	W	PRICE
1N746	3.3	400mm	4/1.00	1N4005	600 PIV	1 AMP	10/1.00
1N751A	5.1	400m	4/1.00	1N4006	800 PIV	1 AMP	10/1.00
1N752	5.6	400m	4/1.00	1N4007	1000 PIV	1 AMP	10/1.00
1N753	6.2	400m	4/1.00	1N3600	50	200m	6/1.00
1N754	6.8	400m	4/1.00	1N4148	75	10m	15/1.00
1N959	8.2	400m	8/1.00	1N4154	35	10m	12/1.00
1N965B	15	400m	4/1.00	1N4305	75	25m	20/1.00
1N5232	5.6	500m	28	1N4734	5.6	1w	28
1N5234	6.2	500m	28	1N4735	6.2	1w	28
1N5235	6.8	500m	28	1N4736	6.8	1w	28
1N5236	7.5	500m	28	1N4738	8.2	1w	28
1N456	25	40m	6/1.00	1N4742	12	1w	28
1N458	150	7m	6/1.00	1N4744	15	1w	28
1N485A	180	10m	6/1.00	1N1183	50 PIV	35 AMP	1.60
1N4001	50 PIV	1 AMP	12/1.00	1N1184	100 PIV	35 AMP	1.70
1N4002	100 PIV	1 AMP	12/1.00	1N1185	150 PIV	35 AMP	1.50
1N4003	200 PIV	1 AMP	12/1.00	1N1186	200 PIV	35 AMP	1.80
1N4004	400 PIV	1 AMP	12/1.00	1N1188	400 PIV	35 AMP	3.00

SCR AND FW BRIDGE RECTIFIERS

C36D	15A @ 400V	SCR	$1.95
C38M	35A @ 200V	SCR	1.95
2N2328	1.6A @ 200V	SCR	50
MDA 980-1	12A @ 50V	FW BRIDGE REC.	1.95
MDA 980-3	12A @ 200V	FW BRIDGE REC.	1.95

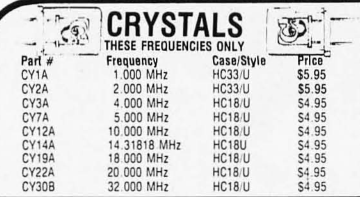
TRANSISTORS

MPS A05	5/$1.00			PN4249	4/$1.00
MPS A06	5/$1.00			PN4250	4/$1.00
2N2219A	3/$1.00	PN3567	3/$1.00	2N4400	4/$1.00
2N2221	4/$1.00	PN3568	4/$1.00	2N4401	4/$1.00
2N2222A	5/$1.00	PN3569	4/$1.00	2N4402	4/$1.00
2N2369	5/$1.00	2N3704	5/$1.00	2N4403	4/$1.00
2N2369A	4/$1.00	2N3705	5/$1.00	2N4409	5/$1.00
2N2484	4/$1.00	2N3706	5/$1.00	2N5086	4/$1.00
2N2906A	4/$1.00	2N3707	5/$1.00	2N5087	4/$1.00
2N2907A	5/$1.00	2N3711	5/$1.00	2N5088	4/$1.00
2N2925	5/$1.00	2N3724	$ .65	2N5089	4/$1.00
2N3053	2/$1.00	2N3725	$1.00	2N5129	5/$1.00
2N3055	$ .89	2N3903	5/$1.00	2N5138	5/$1.00
MJE3055	$1.00	2N3904	4/$1.00	2N5139	5/$1.00
MJE2955	$1.25	2N3905	4/$1.00	2N5209	5/$1.00
2N3392	5/$1.00	2N3906	4/$1.00	2N5951	5/$1.00
2N3398	5/$1.00	2N4013	3/$1.00	C106B1SCR	2/$1.00
		2N4014	3/$1.00	2N5432	$2.00
		2N4123	10/$1.00		

CAPACITOR CORNER

50 VOLT CERAMIC DISC CAPACITORS

	1-9	10-49	50-100		1-9	10-49	50-100
10 pf	.05	.04	.03	.001µF	.05	.04	.035
22 pf	.05	.04	.03	.0047µF	.05	.04	.035
47 pf	.05	.04	.03	.01µF	.05	.04	.035
100 pf	.05	.04	.03	.022µF	.06	.05	.04
220 pf	.05	.04	.03	.047µF	.06	.05	.04
470 pf	.05	.04	.035	.1µF	.12	.09	.075
100 VOLT MYLAR FILM CAPACITORS							
.001mf	.12	.10	.07	.022mf	.13	.11	.08
.0022	.12	.10	.07	.047mf	.21	.17	.13
.0047mf	.12	.10	.07	.1mf	.27	.23	.17
.01mf	.12	.10	.07	.22mf	.33	.27	.22
+20% DIPPED TANTALUMS (SOLID) CAPACITORS							
.1/35V	.28	.23	.17	1.5/35V	.30	.26	.21
.15/35V	.28	.23	.17	2.2/25V	.31	.27	.22
.22/35V	.28	.23	.17	3.3/25V	.31	.27	.22
.33/35V	.28	.23	.17	4.7/25V	.32	.28	.23
.47/35V	.28	.23	.17	6.8/25V	.36	.31	.25
.68/35V	.28	.23	.17	10/25V	.40	.35	.29
1.0/35V	.28	.23	.17	15/25V	.63	.50	.40
MINIATURE ALUMINUM ELECTROLYTIC CAPACITORS							
Axial Lead				Radial Lead			
.47/50V	.15	.13	.10	.47/25V	.15	.13	.10
1.0/50V	.16	.14	.11	.47/50V	.16	.14	.11
3.3/50V	.15	.13	.10	1.0/16V	.15	.13	.10
4.7/25V	.16	.14	.12	1.0/25V	.16	.14	.11
10/25V	.15	.13	.10	1.0/50V	.16	.14	.11
10/50V	.16	.14	.12	4.7/16V	.15	.13	.10
22/25V	.17	.15	.12	4.7/25V	.15	.13	.10
22/50V	.24	.20	.18	4.7/50V	.16	.14	.11
47/25V	.19	.17	.15	10/16V	.14	.12	.09
47/50V	.25	.21	.19	10/25V	.15	.13	.10
100/25V	.24	.20	.18	10/50V	.16	.14	.12
100/50V	.35	.30	.28	47/50V	.24	.21	.19
220/25V	.32	.28	.25	100/16V	.19	.15	.14
220/50V	.45	.41	.38	100/25V	.24	.20	.18
470/25V	.33	.29	.27	100/50V	.35	.30	.28
1000/16V	.55	.50	.45	220/16V	.33	.23	.17
2200/16V	.70	.62	.55	470/25V	.31	.28	.26



CRYSTALS

THESE FREQUENCIES ONLY

Part #	Frequency	Case/Style	Price
CY1A	1.000 MHz	HC33/U	$5.95
CY2A	2.000 MHz	HC33/U	$5.95
CY3A	4.000 MHz	HC18/U	$4.95
CY7A	5.000 MHz	HC18/U	$4.95
CY12A	10.000 MHz	HC18/U	$4.95
CY14A	14.31818 MHz	HC18U	$4.95
CY19A	18.000 MHz	HC18/U	$4.95
CY22A	20.000 MHz	HC18/U	$4.95
CY30B	32.000 MHz	HC18/U	$4.95

XR-2260KB Kit $27.95 **XR-2206KA Kit $17.95**

EXAR

WAVEFORM GENERATORS

XR-205	$8.40
XR-2206CP	4.49
XR-2207CP	3.85

STEREO DECODERS

XR-1310CP	$3.20
XR-1310EP	3.20
XR-1800P	3.20
XR-2567	2.99

MISCELLANEOUS

XR-2211CP	$6.70
XR-4136	.99
XR-1468	3.85
XR-1488	5.80
XR-1489	4.80
XR-2208	5.20

TIMERS

XR-555CP	$ .49
XR-320P	1.55
XR-556CP	1.85
XR-2556CP	3.20
XR-2240CP	3.25

PHASE LOCKED LOOPS

XR-210	5.20
XR-215	6.60
XR-567CP	1.95
XR-567CT	1.70

CONNECTORS

PRINTED CIRCUIT EDGE-CARD

.156 Spacing-Tin-Double Read-Out
Bifurcated Contacts — Fits .054 to .070 P.C. Cards

15/30	PINS (Solder Eyelet)	$1.95
18/36	PINS (Solder Eyelet)	$2.49
22/44	PINS (Solder Eyelet)	$2.95
50/100 (.100 Spacing)	PINS (Solder Eyelet)	$6.95

25 PIN-D SUBMINIATURE

DB25	PLUG	$3.25
DB25	SOCKET	$4.95



3½ DIGIT DVM KI.



This 0-2 VDC .05 per cent digital voltmeter features the Motorola 3½ digit DVM chip set. It has a .4" LED display and operates from a single +5V power supply. The unit is provided complete with an injection molded black plastic case complete with Bezel. An optional power supply is available which fits into the same case as the 0-2V DVM allowing 117 VAC operation.

A. 0-2V DVM with Case $49.95
B. 5V Power Supply $14.95

VECTOR WIRING PENCIL

Vector Wiring Pencil P173 consists of a hand held featherweight (under one ounce) tool which is used to guide and wrap insulated wire, fed off a self-contained replaceable bobbin, onto component leads or terminals installed on pre-punched "P" Pattern Vectorbord. Connections between the wrapped wire and component leads, pads or terminals are made by soldering. Complete with 250 FT of red wire **$9.95**

REPLACEMENT WIRE — BOBBINS FOR WIRING PENCIL

W36-3-A-Pkg. 3	250 ft. 36 AWG GREEN	$2.40
W36-3-B-Pkg. 3	250 ft. 36 AWG RED	$2.40
W36-3-C-Pkg. 3	250 ft. 36 AWG CLEAR	$2.40
W36-3-D-Pkg. 3	250 ft. 36 AWG BLUE	$2.40

1/16 VECTOR BOARD

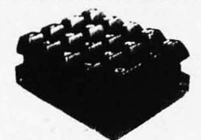
	0.1" Hole Spacing Part No.	P-Pattern L	P-Pattern W	Price 1	Price 2-Up
PHENOLIC	64P44 062XXXP	4.50	6.50	1.72	1.54
	169P44 02XXXP	4.50	17.00	3.69	3.32
EPOXY GLASS	64P44 062	4.50	6.50	2.07	1.86
	84P44 062	4.50	8.50	2.56	2.31
	169P44 062	4.50	17.00	5.04	4.53
	169P84 062	8.50	17.00	9.23	8.26
EPOXY GLASS COPPER CLAD	169P44 062C1	4.50	17.00	6.80	6.12

HEAT SINKS



205-CB	Beryllium Copper Heat Sink with Black Finish for TO-5	$ .25
291-.36H	Aluminum Heat Sink for TO-220 Transistors & Regulators	$ .25
680-.75A	Black Anodized Aluminum	$1.60

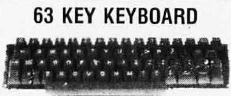
HEXADECIMAL ENCODER 19-KEY PAD



- 1 - 0
- ABCDEF
- Return Key
- Optional Key (Period)
- — Key

$10.95 each

63 KEY KEYBOARD

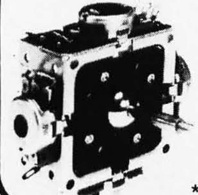


This keyboard features 63 unencoded SPST keys, unattached to any kind of P.C.B. A very solid molded plastic 13" x 4" base suits most applications

$19.95

HD0165 16 LINE TO FOUR BIT PARALLEL KEYBOARD ENCODER $7.95

JOYSTICK



These joysticks feature four * potentiometers, that vary resistance proportional to the angle of the stick. Sturdy metal construction with plastics components only at the movable joint. Perfect for electronic games and instrumentation.

***5K Pots $6.95**
***100K Pots $7.95**

MICROPROCESSOR COMPONENTS

8080A	CPU	$19.95	8228	System Controller - Bus Driver	$10.95
8212	8 Bit Input/Output	4.95	MC6800L	8 Bit MPU	35.00
8214	Priority Interrupt Control	15.95	MC6820L	Periph. Interface Adapter	15.00
8216	Bi-Directional Bus Driver	6.95	MC6810AP1	128 x 8 Static RAM	6.00
8224	Clock Generator/Driver	10.95	MC6830L7	1024 x 8 Bit ROM	18.00

CPU'S

8080	Super 8008	24.95
8080A	Super 8008	19.95

SR'S

2504	1024 Dynamic	$ 3.95
2518	Hex 32 BIT	7.00
2519	Hex 40 BIT	4.00
2524	512 Dynamic	2.49
2525	1024 Dynamic	6.00
2527	Dual 256 BIT	3.95
2529	Dual 512 BIT	4.00
2532	Quad 80 BIT	3.95
2533	1024 Static	7.95
3341	Fifo	6.95
74LS670	16 x 4 Reg	3.95

UART'S

AY-5-1013	30K Baud	$5.95

ROM'S

2513	Char. Gen.	$ 9.95
2516	Char. Gen.	10.95
74S387	1024-Bit Programmable	1.95

RAM'S

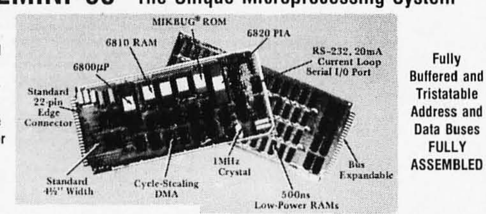
1101	256 x 1	Static	$ 1.49
2101	256 x 4	Static	5.95
2102	1024 x 1	Static	1.75
2107/5280	4096 x 1	Dynamic	4.95
2111	256 x 4	Static	6.95
7489	16 x 4	Static	2.49
8101	256 x 4	Static	6.95
8111	256 x 4	Static	6.95
8599	16 x 4	Static	3.49
91L02	1024 x 1	Static	2.25
74200	256 x 1	Static	6.95
93421	256 x 1	Static	2.95
MM5262	2K x 1	Dynamic	2 for 1.00

PROMS

1702A	2048	Famos	$ 9.95
5203	2048	Famos	14.95
82S23	32 x 8	Open C	5.00
82S123	32 x 8	Tristate	5.00
74S287	1024	Static	7.95
3601	256 x 4	Fast	3.95
2708	8K	Eprom	34.95
6301-1	1024	Tri-State Bipolar	3.49
6330-1	256	Open Collector Bipolar	2.95
6331-1	256	Tri-State Bipolar	2.95

GEMINI-68 The Unique Microprocessing System

Standard 4½" Boards - Dual 22 pin edge Connector NOT A KIT



Fully Buffered and Tristatable Address and Data Buses FULLY ASSEMBLED

STAND ALONE CPU BOARD — Has 384 bytes of RAM on board, serial I/O (RS-232 and 20 ma current loop, cycle stealing direct memory access (DMA), built in soft-ware — selectable echo-back capability.
Part # SA-CPU Board **$279.95**

CPU BOARD — Same as above but only has 128 bytes of RAM on board-used with 8K RAM board listed below.
Part # Gemini 68 CPU Board **$259.95**

8K RAM BOARD — Uses low power static RAMS, 500ns cycle time, 1.5 Amps Max
Part # Gemini 68 RAM Board **$269.95**

8K EPROM BOARD — Uses 5204 EPROMS by AMI or NATIONAL. Shipped with all decode and miscellaneous IC's, except the 5204 EPROMS
Part # Gemini 68 EPROM Board **$ 89.95**

Allow approximately four weeks per delivery

SPECIAL REQUESTED ITEMS

MC3061P	3.50	CD4508	6.75	82S115	25.00	3341	6.95
MC4016P (74416)	7.50	CD4515	6.50	5841	9.95	9368	3.95
MC14583	3.50	CD4520	2.70	MK50240	17.50	MC1408L7	9.95
MC14562	14.50	MCM6571	17.50	11C90	19.95	LD110/LD111	25.00/set
CD4059	9.95	MCM6574	17.50	DS0026CH	3.75	AY-5-9100	17.50 ea.
CD4070	.95	MCM6575	17.50	TIL308	10.50	95H90	13.95

THE BUGBOOKS— Integrated Circuits Applications & Experiments



BUGBOOK I & II - Basic concepts of TTL Logic — over 90 experiments **$17.00/set**

BUGBOOK IIa - Introduces UART — recommended for RTTY enthusiast **$5.00/book**

BUGBOOK III - Explores 8080 chip — introduces Mark 80 Microcomputer **$15.00/book**

555 TIMER APPLICATIONS SOURCEBOOK WITH EXPERIMENTS — over 100 design techniques **$6.95/book**

Introductory Offer — all 5 books (worth $43.95)
SPECIAL - $39.95

CONTINENTAL SPECIALTIES

PROTO BOARD 6
$15.95
(6" long X 4" wide)

Other CS Proto Boards

PB100 - 4.5" x 6"	$ 19.95
PB101 - 5.8" x 4.5"	29.95
PB102 - 7" x 4.5"	39.95
PB103 - 9" x 6"	59.95
PB104 - 9.5" x 8"	79.95
PB203 - 9.75 x 6½ x 2¾	75.00
PB203A - 9.75 x 6½ x 2¾ (includes power supply)	120.00

Logic Monitor **$84.95**
for DTL, HTL, TTL or CMOS Devices

PROTO CLIPS

14 PIN	$4.50
16 PIN	4.75
24 PIN	8.50

DESIGN MATES

DM1 - Circuit Designer 54.95
DM2 - Function Generator 69.95
DM3 - RC Bridge 59.95

QT PROTO STRIPS

QT type	#holes	price
QT-59S	590	12.50
QT-59B	bus strip	2.50
QT-47S	470	10.00
QT-47B	bus strip	2.25
QT-35S	350	8.50
QT-35B	bus strip	2.00
QT-18S	180	4.75
QT-12S	120	3.75
QT-8S	80	3.25
QT-7S	70	3.00

Experimentor 300	$ 9.95
Experimentor 600	$10.95

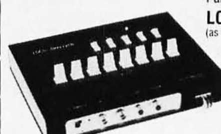
$5.00 Minimum Order — U.S. Funds Only
California Residents — Add 6% Sales Tax

Spec Sheets - 25¢ — Send 35¢ Stamp for 1977A Catalog
Dealer Discount Available — Request Pricing





1021-A HOWARD AVE., SAN CARLOS, CA. 94070
PHONE ORDERS WELCOME — (415) 592-8097
All Advertised Prices Good Thru April



Panatronics
LOGIC ANALYZER KIT
(as shown in Feb. Popular Electronics)

Troubleshoot any type of digital system including Microcomputers.

(includes IC, power supply, PC Board, case and manual)

LA100 KIT $189.00

TV ANTENNA SWITCH

- More than 60 db isolation
- Switches 300 ohm to 300 ohm
- Switches 75 ohm to 300 ohm
- RF coax connecting cable (16ft.) with RCA phono plyes

$6.95

CB MOBiLE MONiTOR

"Hear the highway action"

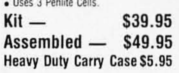


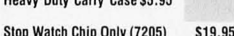
- Off-on volume control
- Internal mount antenna
- 9-volt battery powered
- Portable — attaches to visor
- 23-Channel operation
- Long range reception (Battery not included)

Actual Size - 4" x 3" x 1½ **$9.95**

DIGITAL STOPWATCH

- Bright 6 Digit LED Display
- Times to 59 minutes 59.59 seconds
- Crystal Controlled Time Base
- Three Stopwatches in One: Times Single Event — Split & Taylor
- Size 4.5" x 2.15" x .90" (4½ ounces)
- Uses 3 Penlite Cells.









Kit — $39.95
Assembled — $49.95
Heavy Duty Carry Case $5.95

Stop Watch Chip Only (7205) $19.95

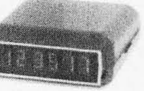
Timeband™

A Trademark of Fairchild Camera and Instrument Corporation

DIGITAL ALARM CLOCK $16.95

- 24-Hour Alarm
- "DOZE" Button
- 100% Solid State
- Large Red Led Display (.8" high)
- AM/PM Indicator
- Seconds Display at touch of button
- SPECIFY BLACK OR IVORY

DIGITAL QUARTZ CAR CLOCK



Complete kit from mounting bracket of the injection molded case down to the three conductor power cord and all components including MM5314 clock chip. Features quartz accuracy of .01%, six digit, .35" high LED display, and P.C. Boards. Works on any 12 volt system — motorcycles, boats, vans, motorhomes, autos, and trucks.

DIMENSIONS 4½" x 4" x 2"
12 or 24 HOUR MODE

Kit: $29.95
Assembled: $39.95

CASE ONLY (includes hardware, mounting bracket and bezel) **$6.50**

JE700 CLOCK



12 or 24 Hour

115 VAC

The JE700 is a low cost digital clock, but is a very high quality unit. The unit features a simulated walnut case with dimensions of 6" x 2½" x 1". It utilizes a MAN72 high brightness readout, and the MM5314 clock chip.

$17.95

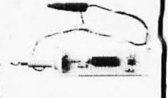
DIGITAL CLOCK KIT — 3½ INCH DIGITS

4 DIGIT KIT $49.95 **4 DIGIT ASSEMBLED $59.95**
6 DIGIT KIT $69.95 **6 DIGIT ASSEMBLED $79.95**

This clock features big 3½" high digits for viewing in offices, auditoriums, etc. Each digit is formed by 31 bright 0.2" LED's. The clock operates from 117 VAC, has either 12 or 24 hr. operation. The 6 digit version is 27" x 3½" x 1½" and the 4 digit is 18" x 3½" x 1½". Kits come complete with all components, case and transformer.

Specify 12 or 24 Hour When Ordering

JE803 PROBE



The Logic Probe is a unit which is for the most part indespensible in trouble shooting logic families TTL, DTL, RTL, CMOS. It derives the power it needs to operate directly off of the circuit under test, drawing a scant 10 mA max. It uses a MAN3 readout to indicate any of the following states by these symbols (H) - 1 (LOW) - o (PULSE) - P. The Probe can detect high frequency pulses to 45 MHz. It can't be used at MOS levels or circuit damage will result.

$9.95 Per Kit
printed circuit board

T²L 5V 1A Supply



This is a standard TTL power supply using the well known LM309K regulator IC to provide a solid 1 AMP of current at 5 volts. We try to make things easy for you by providing everything you need in one package, including the hardware for only **$9.95 Per Kit**

computer display terminal

This display terminal has an integral controller, B/W cathode ray tube and keyboard. The system has a serial I/O interface for communication and an I/O interface for a printer

DISPLAY (P/N 4802-1095-501) FEATURES:
- 17" B/W CRT
- 41 lines of data
- 52 characters per line
- Characters are generated by a diode matrix "graphic" technique
- 21 special push-buttons wired for a program call up
- Brightness Control
- Self-contained power supply

KEYBOARD (P/N 4802-1115-501) FEATURES:
- Reed switch technology
- 54 data keys
- 28 special keys detachable with cable

LOGIC UNIT (P/N 4802-1157-502) FEATURES:
- 1024 by 6 bit core memory
- Printer I/O interface
- Communication I/O interface

POWER: 115V, 50/60 Hz, 500 Watts

WEIGHT: 210 lbs. (including logic unit, keyboard, display and cables.)

FOB LYNN MASS (you pay shipping)
Check with order please.



External logic & power pack not shown.

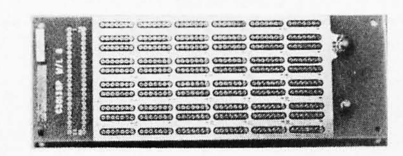
$180.00

"AS IS"

4 way cursor control, graphics display.

The story: These are unused terminals made for airport ticketing & seat assignment. After several years of storage they require tinkering to make operable. We have some hints printed such as cleaning PC fingers. One of our customers has this tied into his KIM-1, another has his running with his IMSAI. We have data on this. Should be useable on most common computers. A hell of a deal and all for a paltry $180.00. Don't be left out as many were on our past VIATRON deal. Sold "as is" all sales final.

WITH COMPLETE DOCUMENTATION

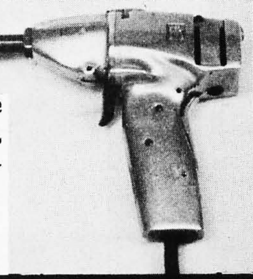


Here is a real deal in a PC module 6x5 sockets (30). List price over $50 each, most by AUGAT. Some pre-wired. New, unused boxed, 14 or 16 pin, 5 x 6 sockets. $15.00 each or 2 for $25, state your choice 14 or 16 pin.

WIRE WRAP GUNS

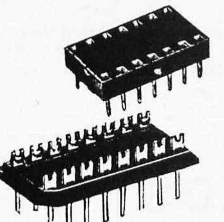
Used wire wrap guns, released due to factory closure. Various mfgrs, some Ingersol Rand, electric or air. No collets. State choice.
Cost over $100.00 each

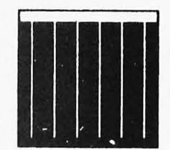
Our price only $15.00 each.

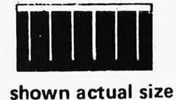


IC Sockets, while they last . . .

8 Pin	10/$1.00
14 Pin	10/$1.25
16 Pin	10/$1.50
18 Pin	10/$1.75
14 Pin IC connector	10/$1.25







shown actual size

SOLAR CELLS

Designed for the space program, these are the highly efficient silicon high output cells. Used for powering equipment, charging batteries. Made by Ion Physics Corp. Each with spec sheet.
Size .394 x .788" 65 mA, .43 V $1.25 12/$12.00
Size .788 x .788" 125 mA, .43 V $1.60 12/$15.00

Meshna

Please add shipping cost on above. Minimum order $10
FREE CATALOG SP-9 NOW READY
P.O. Box 62, E. Lynn, Massachusetts 01904

S.D. SALES CO. P.O. BOX 28810 – B DALLAS, TEXAS 75228

JUMBO LED CAR CLOCK



$16.95 KIT

Alarm Option – $1.50
AC XFMR – $1.50

THE HOTTEST SELLING KIT WE EVER PRODUCED!

You requested it! Our first D.C. operated clock kit. Professionally engineered from scratch. Not a makeshift kluge as sold by others. Features:

A. Bowmar Jumbo –.5 inch LED array.
B. MOSTEK – 50250 – Super Clock Chip.
C. On board precision crystal time base.
D. 12 or 24 Hr. Real Time Format.
E. Perfect for cars, boats, vans, etc.
F. P.C. Board and all parts (less case) included.

50,000 SATISFIED CLOCK KIT CUSTOMERS CANNOT BE WRONG!

THIS MONTH'S SPECIALS

AMD – 8080A	$14.95
Z-80 CPU	49.95
82S129 1K PROM	2.50

1702A 2K EPROM

We tell it like it is. We could have said these were factory new, but here is the straight scoop. We bought a load of new computer gear that contained a quantity of 1702 A's in sockets. We carefully removed the parts, verified their quality, and are offering them on one heck of a deal. First come, first served. Satisfaction guaranteed! U.V. Eraseable. **$6.95 ea. 4/$25**

UP YOUR COMPUTER!
21L02-1 1K LOW POWER 500 NS STATIC RAM Time is of the essence!

And so is power. Not only are our RAM's faster than a speeding bullet but they are now very low power. We are pleased to offer prime new 21L02–1 low power and super fast RAM's. Allows you to STRETCH your power supply farther and at the same time keep the wait light off.

8 for $12.95

60 HZ CRYSTAL TIME BASE
S.D. SALES EXCLUSIVE!

$5.95 ea. 2/$10.00

KIT FEATURES:

A. 60HZ output with accuracy comparable to a digital watch.
B. Directly interfaces with all MOS clock chips.
C. Super low power consumption (1.5 MA typ.)
D. Uses latest MOS 17 stage divider IC.
E. Eliminates forever the problem of AC line glitches.
F. Perfect for cars, boats, campers, or even for portable clocks at ham field days.
G. Small size; can be used in existing enclosures.

Kit includes Crystal, Driver IC, PC board, plus all necessary parts and specs. At last count – over 20,000 sold!

S.D. SALES EXCLUSIVE
$12.95 MOS 6 DIGIT UP–DOWN COUNTER $12.95

40 PIN DIP. Everything you ever wanted in a counter chip. Features: Direct LED segment drive, single power supply (12 VDC TYPE.), six decades up/down, pre-loadable counter, separate pre-loadable compare register with compare output, BCD and seven segment outputs, internal scan oscillator, CMOS compatible, leading zero blanking. 1MHZ. count input frequency. Very limited quantity! WITH DATA SHEET

7400—19c	7411—29c	7451—19c	7490—65c	74153—75c
74LS00—49c	7413—50c	7453—19c	74LS90—95c	74154-1.00
7402—19c	7416—69c	7473—39c	7492—75c	74157—75c
74LS02—49c	7420—19c	7474—35c	7493—69c	74161—95c
7404—19c	7430—19c	74LS74-59c	7495—75c	74164-1.10
74L04—29c	7432—34c	7475—69c	7496—89c	74165-1.10
74S04—44c	7437—39c	7476—35c	74121-38c	74174—95c
74LS04-49c	7438—39c	7480—49c	74123-65c	74181-2.50
7406—29c	7440—19c	7483—95c	74132-1.70	74191-1.25
7408—19c	7447—85c	7485—95c	74S138-1.95	74192-1.25
7410—19c	7448—85c	7486—45c	74141-75c	74193-1.00
				74195—69c

TTL INTEGRATED CIRCUITS

1000 MFD Filter Caps



Rated 35 WVDC Upright style with PC leads. Most popular value for hobbyists. Compare at up to $1.19 ea. from franchise type electronic parts stores. S.D. Special 4/$1.

Slide Switch Assortment



Our best seller. Includes miniature and standard sizes; single and multiposition units. All new, first name brand. Try one package and you'll reorder more! Special **12/$1.00**

RESISTOR ASSORTMENT

1/4W 5% & 10% PC leads. A good mix of values. **200/$2.**

P.C. LEAD DIODES

1N4148/1N914 100/$2.00
1N4002-1A. 100 PIV 40/$1.

HEAVY DUTY Full Wave Bridge

25 AMP 50 PIV
$1.25

Disc Cap Assortment

PC leads. At least 10 different values. Includes .001, .01, .05, plus other standard values. 60/$1.00

SIX DIGIT ALARM CLOCK KIT

$9.95 KIT

P.C. Board – 3.00
AC XFMR – 1.50

Do not confuse with Non-Alarm kits sold by our competition! Eliminate the hassle – avoid the 5314!

We made a fantastic kit even better. Redesigned to take advantage of the latest advances in I.C. clock technology. Features: Litronix Dual ½" displays, Mostek 50250 super clock chip, single I.C. segment driver, SCR digit drivers. Greatly simplified construction. More reliable and easier to build. Kit includes all necessary parts (except case). P.C.B. or XFMR optional. *NEW! WITH JUMBO LED READOUTS!*

Motorola SCR

2N4443. 8 AMP 400 PIV. P.C. Leads 3/$1.

FAIRCHILD - TBA 641

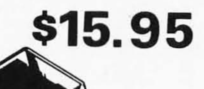
4W. Audio power Amp. Just out! In special heat sink DIP. One super audio IC. $1.50 with data

FND-359 -Led Readout

.4 IN. Common Cathode. High effeciency. Has FND-70 PIN OUT. 59c

COMPUTER POWER SUPPLY

$15.95



A very fortunate purchase. One of the best industrial quality REGULATED supplies we have seen. High performance, small size. Input is 120 VAC 60 HZ. Has the following regulated outputs: **–5VDC@800MA; –15VDC @ 1.25 AMP; –25VDC @ 180 MA.** Sold at a fraction of original cost. Do yourself a favor and order NOW. We expect a quick sellout.

NEXT MONTH:
S.D. will have music for your ears. Watch our ads.

For your Imsai or Altair 8080 Computer:
Z–80 CPU Kit – $149. 4K Low Power Ram Kit – $89.95

OUR CATALOG

is chocked full of rare parts bargains, deals, RAM or CPU kits, plus much more. Yours FREE!

PRICES SHOWN SUBJECT TO CHANGE WITHOUT NOTICE.

Terms: Money back guarantee. No COD. Texas residents add 5% sales tax. Add 5% of order for postage & handling. Orders under $10. add 75c. Foreign orders: US funds only!

Call your Bankamericard or Master Charge order in on our continental United States toll free Watts:
1–800–527-3460
Texas Residents Call Collect:
214/271-0022

Special Thanks to: Dennis, Fred, Abe, Bill, Sam, Hal, Tom, Alex, John, Ely, and Larry

S.D. SALES CO.
P. O. BOX 28810 B
Dallas, Texas 75228

ORDERS OVER $15.00 – CHOOSE $1.00 FREE MERCHANDISE

SOLID STATE SALES



Features

INTEGRATED CIRCUITS

HARDWARE + P.C. SOCKETS

SILICON POWER RECTIFIERS

ZENERS

TRIACS

SCRS

LIGHT EMITTING DIODES

What more could you ask for ?

We've got it.

2708-PROM	$55.00
2522 STATIC SHIFT RE6.	$1.95
2518-HEX 32 BIT SR	$3.50
2102-1 1024 BT RAM	$1.80
5280-4K DYNAMIC RAM	$10.50
5202A UV PROM	$10.50
MM5203 UV PROM	$10.50
1702A UV PROM	$10.75
5204-4K PROM	$18.95

MINIATURE MULTI-TURN TRIM POTS 100, 500, 2K, 10K, 100K, 200K, 1 Meg. $.75 each . . . 3/$2.00

MULTI-TURN TRIM POTS Similar to Bourns 3010 style 3/16"x5/8"x1-1/4"; 50, 100, 1K, 10K, 50K ohms $1.50 ea. . . . 3/$4.00

LIGHT ACTIVATED SCR's TO-18, 200V 1A . . . $ 1.75

TRANSISTOR SPECIALS

2N3585 NPN Si TO-66	$ .95
2N3772 NPN Si TO-3	$ 1.60
2N456A PNP GE	$ .75
2N4908 PNP Si TO-3	$ 1.00
2N6056 NPN Si TO-3 Darlington	$ 1.70
2N5086 PNP Si TO-92	4/$ 1.00
2N4898 PNP TO-66	$ .60
2N404 PNP GE TO-5	5/$ 1.00
2N3919 NPN Si TO-3 RF	$ 1.50
MPSA 13 NPN Si TO-92	3/$ 1.00
2N3767 NPN Si TO-66	$ .70
2N2222 NPN Si TO-18	5/$ 1.00
2N3055 NPN Si TO-3	$ .80
2N3904 NPN Si TO-92	5/$ 1.00
2N3906 PNP Si TO-92	5/$ 1.00
2N5296 NPN Si TO-220	$ .50
2N6109 PNP Si TO-220	$ .55
2N3638 PNP Si TO-5	5/$ 1.00
2N651 7 NPN TO-92 Si	3/$ 1.00

C/MOS (DIODE CLAMPED)

74C02– .22	4015– .95	4035–1.10
74C10– .22	4016– .40	4042– .78
74C193-1.50	4017–1.05	4047–2.00
4001– .22	4018–1.00	4049– .40
4002– .22	4019– .25	4050– .40
4006–1.20	4020–1.05	4055–1.50
4007– .22	4022– .95	4066– .80
4009– .42	4023– .22	4071– .22
4010– .42	4024– .75	4076– .70
4011– .22	4027– .40	4081– .22
4012– .22	4028– .88	4520–1.15
4013– .40	4029–1.10	
	4030– .22	

MCA-81 OPTICAL LIMIT SWITCH. $1.50

LED READOUTS

FND 500-.5" C.C.	$1.75
HP 7740-.3" C.C.	$1.40
MAN-7-.3" C.A.	$1.25
NS 33-3 dig. array	$1.35
DL 747	$2.50

PRINTED CIRCUIT BOARD

4-1/2"x6-1/2" SINGLE SIDED EPOXY BOARD 1/16" thick, unetched $.50 ea. . . . 5/$2.20

7 WATT LD-65 LASER DIODE IR $8.95

2N 3820 P FET	$ .45
2N 5457 N FET	$ .45
TIS 43 UJT	$ .35
ER 900 TRIGGER DIODES	4/ $1.00
2N 6028 PROG. UJT	$ .65
8 PIN DIP SOCKETS	$ .24
14 PIN DIP SOCKETS	$ .25
16 PIN DIP SOCKETS	$ .28
18 PIN DIP SOCKETS	$ .30
24 PIN DIP SOCKETS	$ .40
40 PIN DIP SOCKETS	$ .60

VERIPAX PC BOARD

This board is a 1/16" single sided paper epoxy board, 4½"x6½" DRILLED and ETCHED which will hold up to 21 single 14 pin IC's or 8, 16, or LSI DIP IC's with busses for power supply connector. . . . $4.00

MV 5691 YELLOW-GREEN BIPOLAR LED	$1.25
FP 100 PHOTO TRANS	$ .50
RED, YELLOW, GREEN OR AMBER LARGE LED's	ea. $ .20
MOLEX PINS	100/$1.00
	1000/$7.50
10 WATT ZENERS 3.9, 4.7, 5.6, 8.2, 18, 22, 100, 150 or 200V	ea. $ .60
1 WATT ZENERS 4.7, 5.6, 10, 12, 15, 18 OR 22V	ea. $ .25
MC6860 MODEM CHIP	$13.00

Silicon Power Rectifiers

PRV	1A	3A	12A	50A	125A
100	.06	.14	.30	.80	3.70
200	.07	.20	.35	1.15	4.25
400	.09	.25	.50	1.40	6.50
600	.11	.30	.70	1.80	8.50
800	.15	.35	.90	2.30	10.50
1000	.20	.45	1.10	2.75	12.50

SILICON SOLAR CELLS

2¼" diameter

.4V at 500 ma. . . . $4.00 ea., 6/$22.50

REGULATED MODULAR POWER SUPPLIES

+ – 15 VDC AT 100ma 115VAC INPUT	$27.95
5VDC AT 1A. 115VAC INPUT	$24.95
12 VDC AT .5 AMP	$24.95
IN 4148 (IN914)	15/$1.00

Send 25¢ for our catalog featuring Transistors and Rectifiers
145 Hampshire St., Cambridge, Mass.

TANTULUM CAPACITORS

.22UF 35V 5/$1.00	4.7UF 35V 4/$1.00
.47UF 35V 5/$1.00	6.8UF 35V 3/$1.00
.68UF 35V 5/$1.00	22UF 35V $ .40
1UF 35V 5/$1.00	33UF 35V $ .40
2.2UF 20V 5/$1.00	30UF 6V 5/$1.00
3.3UF 35V 4/$1.00	100UF 35V $ .50
	150UF 15V $ .50

M7001 ALARM CLOCK CHIP. . . . $5.75

NATIONAL MOS DEVICES

MM1402– 1.75	MM5057– 2.25
MM1403– 1.75	MM5058– 2.75
MM1404– 1.75	MM5060– 2.75
MM5013– 2.50	MM5061– 2.50
MM5016– 2.50	MM5555– 4.75
MM5017– 2.70	MM5556– 4.75
MM5055– 2.25	MM5210– 1.95
MM5056– 2.25	MM5260– 1.75

30 WIRE WRAP WIRE SINGLE STRAND 100 FT. $1.40

TTL IC SERIES

7400– .15	7442– .52	74126– .40
7401– .15	7445– .70	74151– .70
7402– .15	7446– .70	74153– .65
7403– .15	7447– .70	74154–1.10
7404– .20	7448– .70	74155– .70
7405– .20	7450– .20	74157– .70
7406– .25	7472– .33	74161– .85
7407– .25	7473– .35	74164– .95
7408– .25	7474– .35	74165–1.05
7409– .21	7475– .49	74173–1.40
7410– .15	7476– .35	74174– .95
7411– .20	7480– .35	74175– .92
7412– .20	7483– .70	74176– .75
7413– .45	7485– .88	74177– .79
7414– .70	7486– .30	74180 .70
7416– .25	7489–1.85	74181–2.10
7417– .25	7490– .45	74190–1.20
7420– .20	7491– .70	74191–1.20
7425– .28	7492– .50	74192– .85
7426– .25	7493– .45	74193– .85
7427– .30	7494– .70	74194– .85
7430– .20	7495– .70	74195– .75
7432– .25	7496– .70	74196– .88
7437– .25	74107– .32	74257–1.25
7438– .25	74121– .38	74279– .90
7440– .16	74123– .65	75324–1.75
7441– .85	74125– .40	75491– .65
		75492– .65

MINIATURE DIP SWITCHES
CTS-206-4 Four SPST switches in one minidip package. . . . $1.75
CTS-206-8 Eight SPST switches in a 16 pin DIP package. . . . $1.95

AY-5-1013-A30K ser./par., par./ser., universal UART. . . . $6.95

ALCO MINIATURE TOGGLE SWITCHES

MTA 106 SPDT	$1.20
MTA 206 DPDT	$1.70

Full Wave Bridges

PRV	2A	6A	25A
200	.75	1.25	2.00
400	.95	1.50	3.00
600	1.20	1.75	4.00

SANKEN AUDIO POWER AMPS

Si 1010 G 10 WATTS	$ 7.95
Si 1020 G 20 WATTS	$15.95
Si 1050 G 50 WATTS	$27.95

CCD 110 LINEAR 256 XI BIT SELF SCANNING CHARGED COUPLED DEVICE . . . $99.00

CCD 201 – 100 x 100 CHARGE COUPLED DEVICE . . . $135.00

LINEAR CIRCUITS

LM307 – Op. Amp.	$ .30
LM 309K 5V 1A REGULATOR	$ .95
723 – 40 + 40VV REGULATOR	$ .50
301/748-Hi Per. Op. Amp.	$ .31
320T 5,12,15, or 24V NEG REG	$1.25
709C – Op. Amp.	$ .31
741A or 741C OP AMP.	$ .31
710 COMPARATOR	$ .35
CA 3047 Hi Pef. Op. Amp.	$ .95
340T 5, 6, 8, 12, 15, 18, 24V POS REG. TO-220	$1.10
101 OPER. AMP. HI PERFORM.	$ 75
LM 308 Oper. Amp., Low Power	$ .95
747 – DUAL 741	$ .65
556 – DUAL TIMER	$1.00
537 – PRECISION OP. AMP.	$1 70
LM 3900 – QUAD OP. AMP	$ .49
LM 324 – QUAD 741	$1.50
560 – PHASE LOCK LOOP	$2.00
561 – PHASE LOCK LOOP	$2.00
565 – PHASE LOCK LOOP	$1.25
566 FUNCTION GEN.	$1.65
567 – TONE DECODER	$1.50
LM 1310N FM STEREO DEMOD.	$2.75
8038 IC VOLTAGE CONT. OSC.	$3.90
LM 370 – AGC SQUELCH AMP.	$1.15
555 – 2μs – 2 HR. TIMER.	$ .45
553 QUAD TIMER	$2.50
FCD 810 OPTO-ISOLATOR	$ .80
1458 DUAL OP AMP.	$ .60
LM 380 – 2W AUDIO AMP.	$ .95
LM 377 – 2W Stereo Audio Amp.	$2.50
LM 381 – STEREO PREAMP.	$1.50
LM 382 – DUAL AUDIO PREAMP	$1.50
LM 311 – HI PER. COMPARATOR	$ .90
LM 319 – Dual Hi Speed Comp.	$1.25
LM 339 – QUAD COMPARATOR	$1.50

TRIACS / SCR'S

PRV	TRIACS 1A	TRIACS 10A	TRIACS 25A	SCR'S 1.5A	SCR'S 6A	SCR'S 35A
100	.40	.70	1.30	.40	.50	1.20
200	.70	1.10	1.75	.60	.70	1.60
400	1.10	1.60	2.60	1.00	1.20	2.20
600	1.70	2.30	3.60		1.50	3.00

Terms: FOB Cambridge, Mass. Send Check or Money Order. Include Postage, Minimum Order $5.00, COD'S $20.00

SOLID STATE SALES
P.O. BOX 74B
SOMERVILLE, MASS. 02143 TEL. (617) 547-4005

WE SHIP OVER 95% OF OUR ORDERS THE DAY WE RECEIVE THEM

- Accuracy: ±0.05% of Reading ±1 Count
- Two Voltage Ranges: 1.999 V and 199.9 mV
- Up to 25 Conversions/s
- Z_{in} > 1000 M ohm
- Auto-Polarity and Auto-Zero
- Single Positive Voltage Reference
- Standard B-Series CMOS Outputs–Drives One Low Power Schottky Load
- Uses On-Chip System Clock, or External Clock
- Low Power Consumption: 8.0 mW typical @ ±5.0 V
- Wide Supply Range: e.g., ±4.5 V to ±8.0 V

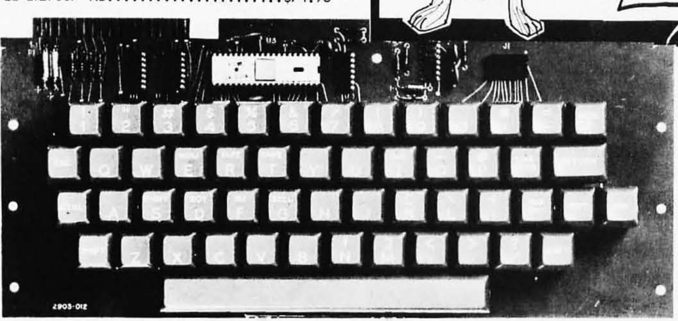
MC14433 SINGLE CHIP 3½ DIGIT A/D
Single chip combines linear and CMOS digital to bring you the simplest yet DVM approach. Requiring only 4 external passive parts, this subsystem gives you: Auto polarity, auto zero, single voltage reference, 8 mW operation, overrange, underrange signals, 25 conversions per second and .05% ± 1 count accuracy! 100 uV resolution. 24 Pin DIP.
MC14433P ....with specs.................$19.55

AMP L'ANNY Says

I'm one year old this month goin' on two – the folks at Tri-Tek will be bringing you full growed values in electronic parts and the latest in books and data.
They've been doin it for six years now---------

ASR-33 KEYBOARD

These are BRAND NEW keyboards from one of the nations largest switch and keyboard manufacturers – not surplus. A special arrangement with the maker allows us to bring you these quality keyboards at less than O.E.M. cost. Available in any quantity, you can design them into your equipment and be sure of continued source. Individual keys can be easily replaced if damaged – you won't wear them out!!
MODEL 2129007 KB......................$74.95

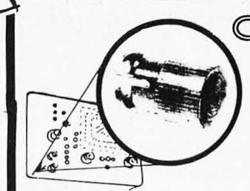


COS-MAC'S BACK!

CDP1802 COSMAC IS BACK IN STOCK !!!!!!!!!
The famous and very popular 1802 is here again after a long dry spell. This is the little 8 bit CMOS microprocessor which sold out on our first offering.
Special AMP'L ANNY birthday price of $29.95 for the month of APRIL only!!
CDP1802.....................................$29.95
CDP1852............ Parallel adapter.......... $15.95
CDP1821SCD................................$24.62
CDP1822SCD................................$25.80
CDP1824CD.................................$13.35

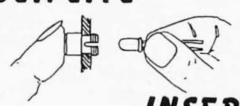
CLIPLITE™

COMBINATION LENS AND MOUNTING DEVICE FOR T 1¾ LED



REQUIRES NO TOOLS

SNAP CLIPLITE



INSERT LED

AVAILABLE IN TRANSPARENT RED · GREEN · AMBER · CLEAR & YELLOW

CLIPLITE
Combination lens and mounting device for T 1-3/4 LED. The CLIPLITE combines the benefits of the present LED display panel mounting methods and eliminates their deficiencies. Requires no special tools and installs in 6 seconds in .250" hole. Simple two-step installation. Just snap CLIPLITE, insert LED. Available in transparent red, green, amber, clear and yellow. Specify colors, any mix.
5/$1.00, 10/$1.90, 20/$3.50, 50/$7.50, 100/$13.50

KEYBOARD ENCODER MM74C922 74C923
CMOS key encoders provide all the necessary logic to fully encode an array of SPST switches. On chip oscillator for scan or use with external clock source. Output is binary and is low power TTL compatible. Keybounce elimination with single capacitor. Memory holds last key depressed. Operates on 3 to 15 volts. 2-key rollover.
MM74C922N......16 key encoder...............$6.35
74C923N 20 key keyboard......................$6.50
Specs for both chips.............................60¢

MM5865 UNIVERSAL TIMER I.C.
A truly universal timer can be used for a stopwatch, kitchen timer, oven timer, event timer/counter, rally timer----. 7 functions, two counters, internal comparators, on chip oscillator. Memory for rally with total elapsed time. Can be cascaded---selectable resolution--count up or down ---selectable modulo for time or event count. Operate on 7 to 20 volts at about 7mA.
MM5865N..................................... $8.75
Specs and applications........................... 60¢

MIL RANGE 5V REGULATOR.
Want a little better performance from your 5V supply? LM140K in TO-3 from contract cancellation brings you a superior part at the price of plastic regulator.
LM140K...$1.95

LM1889 TV VIDEO MODULATOR
The LM1889 is designed to interface audio, color difference, and luminance signals to the antenna terminals of a TV receiver. It consists of a sound subcarrier oscillator, chroma subcarrier oscillator, quadrature chroma modulators, and R.F. oscillators and modulators for two low-VHF channels.
The LM1889 allows video information from VTR's, games, test equipment, or similar sources to be displayed on black and white or color TV receivers.
LM1889 with 16 pages of data $9.95, data only, $1.00



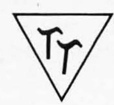
DATA BOOKS BY NATIONAL SEMICONDUCTOR
DIGITAL. Covers TTL, DTL, Tri-State, etc.......$3.95
LINEAR. Covers amplifiers, pre-amps, op-amps... $4.25
LINEAR APPLICATIONS VOLUME I. Dozens of application notes and technical briefs covering the use of op-amps, regulators, phase locked loops and audio amps.................................... $3.25
LINEAR APPLICATIONS VOLUME II. Takes up where Volume I left you--All the latest linear devices. Along with Vol I you have a great source of application data on the most widely used devices as well as new types just appearing.....................$3.25
CMOS Gates, Flip Flops, registers, etc........$3.00
VOLTAGE REGULATORS. A must for anyone making a power supply. Complete theory including transformers, filters, heat sinks, regulators etc........... $3.00
MEMORY. Info on MOS and Bipolar memories, RAMS ROMS, PROMS and decoder/encoders........... $3.95
INTERFACE. Covers peripheral drivers, level translators, line driver/receivers, memory and clock drivers, sense amps, display driver and opto-couplers..... $3.95
SPECIAL FUNCTIONS DATA BOOK. Contains detailed information for specifying and applying special amplifiers, buffers, clock drivers, analog switches and D/A-A/D converter products................. $3.25
AUDIO HANDBOOK Contains detailed discussions, including complete design particulars, covering many areas of audio with real world design examples... $3.25
------------------SPECIAL--------------------
DATA BOOKSHELF. Buy all ten of the National Data books at one time and save $5.10!!!..........$30.00
(All books shipped ppd in US only. Foreign orders please add shipping for 1.5 lbs per book)

30V DIAC. Tiny glass diode for triggering SCR's and triacs. Improve your firing circuit for more reliable and repeatable operations. At a fraction of the O.E.M. price!! DIA-0030.............................10/$1

MINIATURE POWER SUPPLY
5V, .25A, P.C. mounting module supplies logic voltage from 105-125VAC input, 50-400HZ. Ultra-stable and noise free with external trim capability. Made by PMC. (Model MM-5BL)...........$15.95

LINEAR CONTROL DATA BOOK
T.I. linear products data and applications. Price includes shipping charge, U.S. only..................... $4.25
Outside U.S., add postage for 2 lbs.

NEW BOOK FROM NATIONAL
MOS LSI. Giant data book filled with spec and apps on large scale MOS circuits from National Semiconductor Corporation. Price includes shipping in U.S. only...$4.25
Outside U.S., add postage for 2 lbs.



tri-tek, inc.
6522 north 43rd avenue,
glendale, arizona 85301
phone 602 - 931-6949

We pay surface shipping on all orders over $10 US, $15 foreign in US funds. Please add extra for first class or air mail. Excess will be refunded. Orders under $10, add $1 handling. Please add 50¢ insurance. Master charge and Bank America cards welcome, ($20 minimum). Telephone orders may be placed 10AM to 5:30PM daily, Mon thru Fri. Call 602-931-4528. Check reader service card or send stamp for our latest flyers packed with new and surplus electronic components.

Reader Service

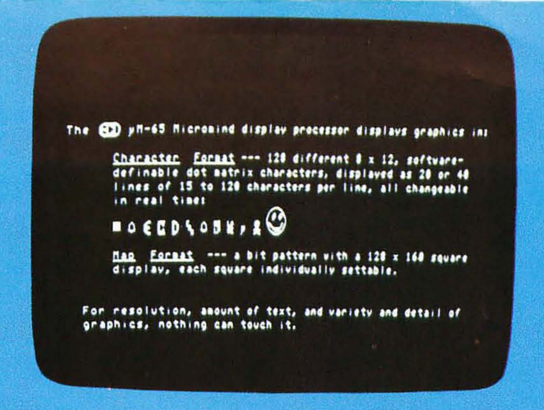
To get further information on the products advertised in BYTE, fill out the reader service card with your name and address. Then circle the appropriate numbers for the advertisers you select from this list. Add a 9 cent stamp to the card, then drop it in the mail. Not only do you gain information, but our advertisers are encouraged to use the marketplace provided by BYTE. This helps us bring you a bigger BYTE.

***Reader service inquiries not solicited. Correspond directly with company.**

BOMB–BYTE's Ongoing Monitor Box

BOMB Exploded

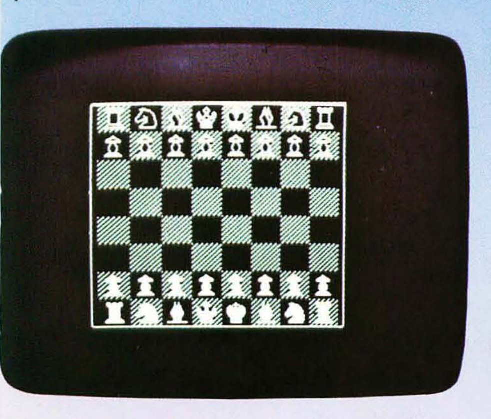
We put BYTE's Ongoing Monitor Box on a postal card starting with the January issue in order to encourage responses. Never did we expect the BOMBslide that resulted. Peter Travisano and Gregory Spitzfaden, who shared the staggering job of totalling the voluminous January responses, have emerged from the countinghouse to announce that David M Allen, who wrote "Saturation Recording's Not All That Hard," and Terry Dollhoff, author of "Making Hash With Tables," were so close in the balloting that each will receive $100 for the top award, instead of our usual first and second place division of BOMB prizes.■



Key Into Maxi-Power @ Micro-Price

Micromind is an incredibly flexible, complete and expandable, hardware/software, general purpose computer system. You won't outgrow it.

Hardware includes an 80 key, software-definable keyboard, I/O interface board, 6500A-series microprocessor (powerful enough for advanced computing), a high-detail graphics and character display processor, power supply, rf modulator, and connections for up to 4 tape recorders plus TV or monitor. An interconnect bus



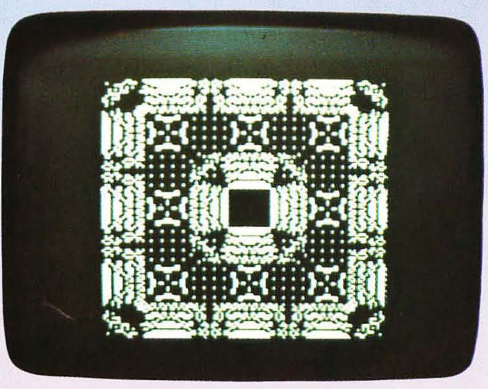
permits 15 additional microprocessors, parallel processing and vastly increased computing power.

System software—including ECD's own notsoBASIC high level language, on advanced error-correcting tape cassettes—provides a word processing editor, a



powerful assembler, a debugger, a file system, graphic routines, and peripheral handlers. We also include dynamic graphic games: Animated Spacewar and Life.

ECD's standard Micromind μM-65 supplies 8K bytes of memory. Additional



32K byte expansion boards and a mapping option give Micromind expandable access to 64 Megabytes. Utilizing software-controlled I/O channels, Micromind's advanced encoding techniques load data from ordinary tape recorders at 3200 bits per second.

Micromind comes to you ready-to-use, factory assembled and fully tested. Among microcomputers, it has the largest memory capacity and the fastest storage. You're looking at the work of the finest display processor on the market. You won't find a microcomputer with a more powerful CPU.

You won't find a computer with a more flexible keyboard. You won't find anything to touch it at **$987.54.**

So, quit the kluge scene and key into Micromind. You'll be a main frame performer, with all the comforts of home. We're not fooling... this is the cat's μ!

ECD CORP.
196 Broadway, Cambridge, Mass. 02139
(617) 661-4400

ECD


Name ______________________
Address ______________________
City/State ______________ Zip ________
☐ Fantastic! Check enclosed: $987.54. Shipping paid by ECD
☐ BankAmericard ☐ Master Charge Mass. Resident add 5% Sales Tax
______________ Expiration Date ________
Signature ______________________
☐ Send me your brochure.
b

Actual unretouched photographs.